FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

Human motion analysis in video sequences for telerehabilitation systems

Ana Clara Ferreira Matos

MASTER THESIS

Integrated Master in Bioengineering

Supervisor: Professor Luís Corte-Real, PhD

Co-Supervisor: Teresa Terroso, PhD

September, 2015

Resumo

As actuais tendências demográficas apontam para um aumento da população envelhecida e de doenças crónicas. Esta situação intensifica a pressão colocada nos sistemas de saúde em fornecer cuidados de qualidade. Muitos dos sintomas de doenças crónicas podem ser aliviados com recurso à reabilitação. O sucesso desta está directamente ligado a começar o tratamento o mais cedo possível e a continuar o tratamento depois da fase crítica. No entanto, devido a uma falta de recursos e pessoal médico especializado muitos doentes não recebem o tratamento adequado. A implementação de um sistema de telereabilitação poderia resolver os problemas mencionados. Todavia, para garantir o sucesso do tratamento de reabilitação em casa é necessário garantir uma monotorização contínua assim como feedback e orientação por parte do pessoal médico. Uma forma de permitir a monitorização mencionada é através do uso de sistemas baseados em marcadores. Apesar da sua exactidão, estes sistemas são relativamente caros e a colocação dos marcadores é demorada, tem que ser realizada por um especialista e pode causar um desconforto considerável. Devido às razões referidas este sistemas são inadequados para serem usados em casa do paciente. Recentemente, sistemas de aquisição 3D baratos, eficientes e fáceis de utilizar têm começado a emergir como uma solução para o problema de rastreamento do movimento no contexto de uma aplicação para reabilitação. Apesar da utilização de sensores activos, tais como luz estruturada, ter sido minuciosamente investigada, a utilização de sensores passivos, tais como uma camera estéreo, tem permanecido menos explorada.

O presente trabalho tem como objectivo explorar a aplicabilidade de um aparelho passivo de aquisição 3D para o rastreamento do movimento num contexto de reabilitação. O sistema proposto utiliza uma camera estéreo para adquirir a informação de cor e profundidade que é entregue ao sistema de rastreamento do esqueleto. A recuperação da posição 3D de articulações predefinidas teve como base relações cinemáticas e comprimentos antropométricos válidos, assim como a consistência temporal. Finalmente, para caracterizar e avaliar os exercícios de reabilitação realizados, foi extraído um conjunto de medidas quantitativas a partir da informação do esqueleto obtida.

Ainda existe espaço para melhorias no sentido de alcançar uma troca adequada entre a exatidão da informação obtida e o tempo de processamento que vai ao encontro das expectativas de aplicações em tempo real. Ainda assim, um estudo de validação realizado com informação de referência fornecida por um sistema baseado em marcadores, revelou que o sistema é capaz de atingir erros dentro da gama de sistemas convencionais activos e da avaliação visual feita por um terapeuta. Os resultados obtidos são promissores e demonstram que a metodologia desenvolvida baseada no uso de um sistema passivo, é capaz de permitir a análise do movimento humano num contexto de reabilitação.

Abstract

The present demographic trends point to an increase in the aged population and in chronic diseases. This situation intensifies the pressure placed on the health care systems in providing quality care. Many of the symptoms of chronic diseases can be alleviated with the appeal of rehabilitation. The success of rehabilitation is directly linked to initiating the treatment as soon as possible and continuing the treatment even after the critic phase. However, due to the lack of resources and medical staff many patients are not receiving the proper treatment. The implementation of a telerehabilitation service could solve the aforementioned problem. Nevertheless, in order to ensure the success of the rehabilitation treatment in the home environment continuous monitoring needs to be guaranteed, as well as feedback and guidance from the medical staff. One way of providing the mentioned monitoring of the patients' performance is through the use of marker based systems. Despite their accuracy, these systems are quite expensive and the setting of the markers is time-consuming, needs to be performed by a specialist and can cause considerable discomfort. For the named reasons these systems are unsuitable to be used in the patients' home. Recently, inexpensive, efficient and easy-to-use 3D acquisition sensors are emerging as a solution for the problem of motion tracking in the context of a rehabilitation application. Even though active sensors, such as structured light, have been thoroughly investigated, the use of passive sensors, such as a stereo camera, has remained unexplored.

The present work aimed to explore the applicability of a passive 3D acquisition device for motion tracking in a rehabilitation context. The proposed system uses a stereo camera to acquire the depth and color information that is passed to the skeleton tracking system. The recovery of the 3D position of predefined joints was based on kinematic relationships and anthropometrically feasible lengths as well as temporal consistency. Finally, a set of quantitative measures were extracted from the obtained skeleton data in order to characterize and evaluate the performed rehabilitation exercises.

There is still room for improvement in order to achieve an adequate trade-off between accuracy and processing time that meets real-time expectations. Nevertheless, a validation study using as ground-truth the data provided by a marker based system revealed that the system was able to reach errors within the range of state-of-the-art active markerless systems and the visual evaluation done by a physical therapist. The obtained results are promising and demonstrate that the developed methodology based on the use of a passive sensor is able to allow the analysis of the human motion for a rehabilitation purpose.

Acknowledgements

This work could not be complete without the acknowledgements to those without whom this could not have been possible. To all my sincere thank you.

First I would like to thank Professor Teresa Terroso for the continuous availability, motivation, wise advices and help throughout this work. For being always available to answer my questions and for believing in my work. To Professor Luís Corte Real for the pertinent input, ideas and questions. To Professor Pedro Carvalho, for the everyday help and the assistance in establishing contacts. I would also like to acknowledge INESC TEC for proving the tools and support needed to the achievement of this work.

Some external people to this project who made a significant contribution: to the LABIOMEP for the availability in providing the location for the acquisition of the marker based software used in the collected dataset, to Pooya Soltani for the patience and help during the acquisition day.

To all my friends, that made these last 5 years unforgettable. For those who made this experience worth remember and to look back with *saudade*. To Ana, Joana, Lia, Maria, Pi and Isa for all the dinners, the gifts and for always being there for each other. A special thanks to Lia for being my model and for being available to perform as many rehabilitation exercises as needed. To Mariana for being my forever lab mate, for always being ready to listen to me. To Bruna, for the wise advices, for the timeless support and for making me see things from a different perspective. To Joana, my timeless roommate. For all the things we've shared during these last 5 years. The greatest years of my life would not have been the same without you.

To Gonçalo, for being my greatest support and my partner. For being always willing to hear my doubts and insecurities, for making an effort to understand my work, for dealing with my bad mood, for always waiting for me just to make my day better (and then making me the best dinners). For the countless hugs and smiles, I cannot thank him enough.

To my parents and my sister. To my parents, for their continuous work in providing me all the tools I needed to be where I am today. For their selflessness and continuous and unconditional support, care and love. To my father, for always pushing me to be better and making me believe that I can achieve all that I set myself to. To my mother, for always taking care of me and for always supporting my aspirations and achievements. To my sister, for looking up to me and for making me set the example. You're my lifelong partner. There are no words that can thank them enough.

Clara Matos

Believe you can and you're halfway there.

Theodore Roosevelt

Contents

List of Figures

List of Tables

Acronyms

2D	Two-dimensional
3D	Three-dimensional
BM	Block Matching
bPSO	Baseline PSO
BSD	Berkeley Software Distribution
CPD	Coherent Point Drift
DoF	Degrees of Freedom
ECG	Electrocardiogram
FPS	Frames per Second
GP	Gaussian Process
GPU	Graphics Processing Unit
HWICP	Hierarchical Weighted Iterative Closest Point
HSV	Hue, Saturation, Value
ICP	Iterative Closest Point
MBS	Marker Based Systems
MH	Make-Human
MI	Mutual Information
MLS	Markerless systems
OpenCV	Open Source Computer Vision Library
PCA	Principal Component Analysis
PCD	Point Cloud Data
PCL	Point Cloud Library
PDA	Principal Direction Analysis
pPSO	Perturbed PSO
PSO	Particle Swarm Optimization
RANSAC	RANdom SAmple Consensus
RDF	Random Decision Forest
RGB-D	Red, Green, Blue - Depth
ROS	Robot Operating System
SAD	Sum of Absolute Differences
SCAPE	Shape Completion and Animation of PEople
SD	Secure Digital
SGBM	Semi-Global Block Matching
ST	Skeleton Tracking
ToF	Time-of-Flight
VAR	Variational Matching
VH	Visual Hull
VR	Virtual Reality

Chapter 1

Introduction

The pressure placed on the health care systems worldwide in their ability to provide quality health care has been increasing as the present demographic trends point to an increase in the aged population and in chronic diseases [1]. In fact, according to the World Health Organization, around 15% of the world's population lives with some form of disability [2]. At the same time, an unprecedented development in the fields of information and communication technologies is being observed [3], with the expansion and availability of internet and broadband connections in most of our homes and workplaces [4]. From the combination of these two situations, the field of telerehabilitation is beginning to arise as a promising solution to deal with the need of better health services [5].

Rehabilitation aims to allow a person who suffered from a motor function disability can achieve the highest possible level of independence. For this reason, the movements performed in a telerehabilitation setting need to be continuously monitored in relation to a correct motion pattern. Hence, detecting and tracking human movements becomes of uttermost importance during the rehabilitation process [6].

1.1 Motivation

The success of the rehabilitation process can only be guaranteed if, during the chronic phase of the recovery process, a continuous activity is maintained [5]. However, as a result of geographical location some patients experience shortage of resources and medical staff and need to travel long distances to receive specialized aid. As a consequence, patients tend to delay or avoid necessary care, with harmful consequences regarding their recovery [7]. By moving the treatment from the hospital facilities to the patient's home, some of the problems resulting from the lack of accessibility [3] could be overcome. Also, the hospitalization time could be reduced and the duration and intensity of the rehabilitation time could be increased resulting in an overall saving of time and resources. However, switching the treatment from the hospital facilities to the patients' home demands the continuous monitoring of the rehabilitation process. The adequate adherence to home exercises involves performing the exercises exactly as prescribed and in the numbers and durations

prescribed. The absence of monitoring results in serious problems: not only the patients can be performing the exercises incorrectly with severe consequences for the rehabilitation process; they can become discouraged due to the lack of interest, guidance and feedback [8]. The aforementioned problems of lack of monitoring and feedback could be solved with the implementation of a telerehabilitation system. This could be achieved simply by using video transmission, such as in the case of telemedicine. However, this solution requires the availability of a therapist to visually monitor and evaluate the patient performing the rehabilitation exercises, which is impracticable given the current shortage of medical staff. By taking advantage of more advanced motion tracking systems, the therapist evaluation would be done based on quantitative measures extracted by the system, removing the need for the continuous visual assessment.

In this environments, human movement tracking systems should be able to generate real-time and accurate data to dynamically represent the position changes of a human body (or portion of it). By providing adequate feedback and guidance these systems could potentiate the proper performance of the rehabilitation exercises and increase the patient accountability and motivation. Also, the identification and correction of errors in the exercises by the clinician could enable the modification of the prescribed exercises and thus minimize costly and unneeded trips to outpatient centers.

The telerehabilitation system could be accomplished using motion-sensor technologies. The current gold standard for motion-sensor technologies are marker-based systems [9]. Although these systems provide high accuracy, they are quite expensive, the markers placement needs to be performed by a specialist, takes a considerable amount of time and can be somehow cumbersome specially for kids and patients with reduced mobility. Also, the analysis must be performed in specialized centers [10]. For the stated reasons, the lack of portability and easiness to use makes them unsuited for a home context.

Recently, the development of efficient, affordable, compact and easy to use three-dimensional (3D) acquisition sensors boosted their application for motion tracking in rehabilitation. The three main technologies currently used are: structured light, such as the Microsoft® Kinect; time-of-flight (ToF); stereo cameras [11]. The first two are more efficient in proving information in real-time. In fact, since its release in 2010, Microsoft® Kinect has been used in a wide variety of motion tracking systems, including for rehabilitation purposes [12, 13, 14]. Nevertheless, passive stereo cameras are able to produce depth maps with higher resolution and provide more information regarding the observed scene. Despite being more computationally heavy, recent developments in passive stereo algorithms are allowing the achievement of real time speeds [15]. The Microsoft® Kinect is prepared to work in the called "fixed-camera-living-room scenario" [16]. In opposition, given the modularity of stereo cameras, they could be used in outdoor scenarios, for example. Also, the stereo camera baseline can be adapted to the acquisition environment and avoid the problem of range resolution given by a fixed baseline. As well, the use of more than two views can enable the acquisition of 360º human bodies, rather than the frontal ones provided by the standard structured light devices. With active systems, the use of more than two views is less reliable. Due to the interference between the projected patterns, each camera is unable to distin-

guish its own pattern. This results in a less consistent depth recovery [17]. The acquisition of more complete human bodies can solve many of the problems related to ambiguities and occlusions that hamper motion tracking systems. The described advantages potentiate the use of stereo cameras in a wider range of scenarios.

In a telerehabilitation system, the information needs to be provided not only in real time, but also with the accuracy needed to produce relevant medical information. With this in mind, a question remains: which is the ideal tradeoff between speed and accuracy of a human motion tracking telerehabilitation oriented system? This question motivates the assessment of wide spread and already existing options and the development of new systems that can potentially produce better results in human motion tracking, keeping in mind the requirements of a home based application.

The use of more adaptable sensors could allow the monitoring of patients in less controlled environments, including outdoors; take advantage of the use of several cameras in order to cover a more extended area and allow the tracking of several users simultaneously. For these reasons, the analysis of the applicability of more adaptable sensors for motion tracking in a context of rehabilitation could potentially open the way for the development of more complete and accurate telerehabilitation systems.

1.2 Objectives

The main purpose of the present study was to develop a system for the detection, representation and tracking of the human body as its parts using video sequences. This should be achieved using a stereo camera in order to explore its applicability in opposition to the commonly used active devices. The developed system should be used to extract clinically significant measures in the context of rehabilitation.

Given the absence of available datasets with annotated ground-truth for motion tracking with a stereo system, this project should include the collection of a dataset contemplating the proposed rehabilitation exercises. Based on the information acquired with the annotated dataset, a preliminary evaluation of the performance of the developed system should be done.

1.3 Contributions

The main contributions of the following thesis are described bellow:

- A new methodology was proposed for the acquisition and enhancement of a 3D human body and its surrounding environment from stereo images, suitable to be used as input for a skeleton tracking system;
- A device agnostic skeleton tracking system was improved and used to extract clinically relevant information in the context of rehabilitation;
- The applicability of the use of a passive device as a motion sensor technology in the context of rehabilitation was explored;

- A database containing the performance of three rehabilitation exercises by both a male and a female was acquired. The database comprises color, depth and skeleton data acquired with the Microsoft® Kinect, stereo images acquired with the Bumblebee®2 stereo camera and ground-truth data provided by a marker based system (Qualysis). According to the author knowledge this is the first created database that includes both RGB-D, stereo and marker based information in a context of motion tracking evaluation for rehabilitation;
- Using as ground-truth the information provided by a marker based system, a preliminary validation study of the developed system was conducted.

1.4 Document Structure

This document is organized in five chapters. The introduction explicits the main motivations for the development of the present study. Chapter 2 presents the state of the art regarding the subject of telerehabilitation, giving special attention to the most recent trends on human 3D reconstruction and motion tracking techniques. Chapter 3 describes the proposed method for the analysis of the human motion in a telerehabilitation context. Chapter 4 presents the obtained results and the correspondent discussion. Finally, the conclusions and some proposed future improvements are provided in Chapter 5.

Chapter 2

Literature Review

This chapter aims to review the main Computer Vision concepts related to the analysis of human motion from a telereahabilitation perspective. Section 2.1 reviews some of the questions involved in the implementation of a telerehabilitation service. Section 2.2 presents the state-of-the-art techniques related to the human body 3D reconstruction from depth data. Section 2.3 explores some of the recent approaches applied to tracking human motion using as input the depth information or the 3D data obtained from reconstruction. Finally, some of the tools and software libraries used to solve Computer Vision and 3D processing problems are described in Section 2.4

2.1 Telerehabilitation

Being a relatively recent area of research [3], the definition of telerehabilitation may suffer slightly changes according to different authors. Winters [18] positioned telerehabilitation between the broader and more developed areas of telemedicine and telehealth and subdivided the emerging area into four categories: teleconsultation, teletherapy, telemonitoring and telehomecare. In teleconsultation a interactive videoconferencing between the patient and a caregiver in a faraway location can allow specialized assistance to the patient [4]. During telehomecare, a caregiver, such as a nurse or a therapist, remotely controls the rehabilitation process of the patient. In telemonitoring, which is one of the telerehabilitation areas with more growing prospect, an evaluation technology is placed within the clients' home. In teletherapy the patient follows a set of therapeutic activities remotely managed by a therapist [4].

Telerehabilitation can be perceived considering intensity and duration (Figure 2.1). Intensity is quantified by the information exchanged and can range from high to low intensity. On the other hand, duration is extended from short to long or even lifetime duration [4]. The majority of the telerehabilitation applications fall in the low intensity – long duration quadrant [4]. However the proposed evaluation fails to mention the mode and speed of data transmission. In the process of telerehabilitation we can consider two types of data transmission: interactive (synchronous) or store-forward (asynchronous). The first involves the simultaneous presence of both the patient and

the caregiver, while in the latter the attendance of the stakeholders does not need to be coincident [18].

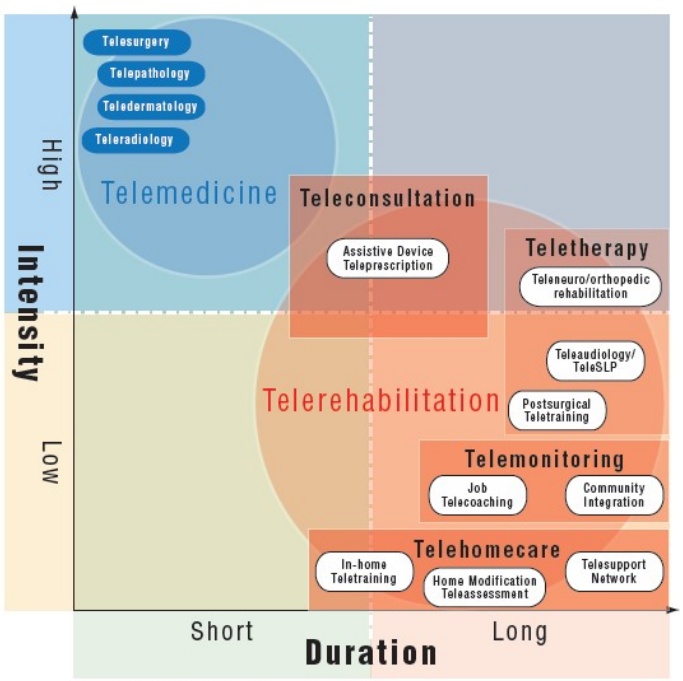

Figure 2.1: Map of telerehabilitation services using the intensity-duration quadrant model. (From [4].)

2.1.1 Benefits and Barriers

Recently the gradual observed shift towards preventive, proactive and continuous care [19] led to an increase in the need of intensive rehabilitation and rehabilitation in chronic phases. These situations combined with the efficiency of the home setup pushed telerehabilitation towards a fast progress. Within the benefits provided by telerehabilitation, the access to health over distance, allowing the access to people in remote locations to otherwise non available health treatments is one of its main premises. Also, a telerehabilitation setting could allow beginning therapy during the acute phase, reduction of hospitalization time, an increase in the intensity and duration of the rehabilitation process, and nonstop monitoring of people at risk resulting in an overall saving of resources and time [5]. Another potential benefit of telerehabilitation is the implementation of the rehabilitation service in the individual's environment. Studies of behavior therapy report that treatments provided in the home setting are more effective than the same therapy delivered in a hospital. Also, therapy at home can increase the ability to perform daily living activities and reduce the incidence of delirium in elderly population [20].

Due to its tender age, research studies under telerehabilitation which report evidences of effectiveness are scarce. In Kairy et al. [1] it is reported that most of the analyzed studies show similar or even better clinical outcomes when compared to the conventional treatments and that there is an emerging evidence for the efficacy of telerehabilitation. Many of the presented studies

often relate to a small number of patients and their design is limited with the absence of controls or randomized data. For this reason, the efficacy of telerehabilitation is not yet fully proved. However, the gathered information [1, 5] points to a trend of positive effect resulting in a promising future for the telerehabilitation field. Ekeland et al. [21] analysed eighty reviews from 2005 to 2010 and concluded that twenty reported effectiveness regarding therapeutic effects, efficiency and technical usability. It is also stated that other nineteen reviews demonstrate promising results, but further research needs to be conducted to establish firm conclusions; other twenty-two mention that evidence is limited and inconsistent.

Many of the benefits promoted by telerehabilitation result in a general saving of time and money [5]. Tousingnant et al. [22] analyzed the costs by considering session duration, therapist salaries, travel time and internet installation and maintenance and concluded that performing the 12 sessions of physiotherapy at home was 17% cheaper. Kortke et al. [23] compared a 3-month trans-telephonic monitoring (a patient activated recording of the heart's rhythm) of ECG signals at home with the 3-week hospitalization offered nowadays in Germany and observed that the home treatment would result in 58% savings. Rojas et al. [24] assessed the cost-effectiveness of telehomecare and concluded that 91% of the studies demonstrated that telehomecare is cost-effective by reducing the use of hospital, improving patient compliance, satisfaction and quality of life [21]. In the published studies there is an overall lack of cost analysis considering not only similar perspectives, but also similar costs and drawing comparisons and conclusions regarding the cost effectiveness of telerehabilitation is very difficult. Likewise, many of the released studies classify telerehabilitation as being cost effective, but don't consider additional costs with equipment, transmission lines, additional personnel and program administration [7]. Regarding cost analysis it is important to have in mind that costs can change over time as the use of technology increases or as the experience of the therapists using the technology grows or that the patients covered by the telerehabilitation service may change due to the saving of costs [1].

As the use of telerehabilitation becomes more widespread, it will continue to face several barriers [7]. One of the main concerns that health professionals demonstrate regarding telerehabilitation systems is the loss of face-to-face contact between the therapist and the patient. Truthfully, the sensory input passively transmitted by the therapist to the patient has been shown to be an important step in the rehabilitation process [5]. However, from the studies analysed by Kairy et al. [1], some demonstrated that the consultation time spent with telerehabilitation systems was similar or longer than the one of face-to-face consults and other [25] reported that the number of contacts between the therapist and the patient was greater, but with shorter consult duration [1]. The second mentioned problem by health professionals is related to the absence of technological background that may hamper the use of the technology. This issue can be prevented by developing user friendly applications and assuring initial training [19]. Another problem that can arise is the patient inclusion at home since sometimes the obligation to visit the rehabilitation center may motivate the patient to go outdoors and participate in the community life and everyday activities which has a strong role in their recovery process. Also, unsupervised physiotherapy may led to the poorly execution of the prescribed exercises with potentially harmful consequences to the

patient [5]. When developing applications for the home environment, the only on-site available assistance is provided by the caregiver and so it is important to provide training not only regarding medical aspects, but also technological ones. Moreover, if a device is placed within the patient's home some requirements must be fulfilled such as simplicity to operate, reliability, high level of fault tolerance and easy troubleshooting and maintenance over distance [19]. Additional obstacles when implementing telerehabilitation systems are related to problems of confidentiality, the adaptation of remote services within the rehabilitation professional's ethics, the absence of standards, guidelines and licensing regulations.

2.1.2 Technology

In last decades, the development of Information and Communication Technologies, such as an increase in computer power and availability of high speed internet, resulted in a wide diversity of technologies available to provide telerehabilitation systems [3, 20, 26]. When creating any telerehabilitation system, universal design standards must be kept in mind in order to allow that the developed application is acessible, efficient, usable and understandable to all [19].

Audio and video can be used in real-time or asynchronously. Real-time videoconferencing can be accomplished by using simple and inexpensive webcams and a common internet connection [26]. Teleconsultation falls within this category and can be used to provide face-to-face contact between a patient in a remote location and a health professional [4]. Vsee® (Figure 2.2) is a videoconferencing and collaboration service that provides an add-on called the Vsee® OneClick that allows online appointments between patients and doctors. An asynchronous use of the application can be accomplished by collecting data and later forwarding it to a clinician to review using email, Bluetooth, or other electronic format [20].

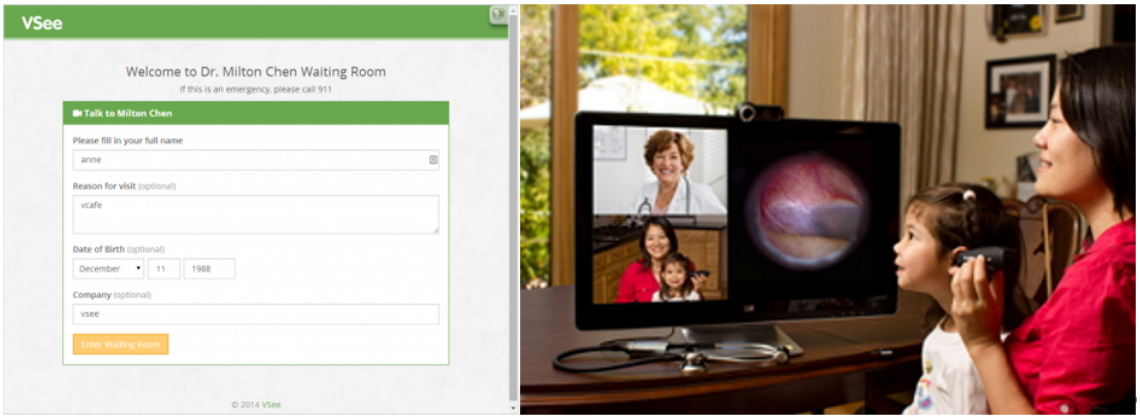

Figure 2.2: Vsee® OneClick application[1].

Sensor based technology gets advantage of a set of apparatus that allow the acquisition and assessment of information regarding the patient physical performance and condition, such as blood pressure, body temperature, heart rate, muscle and brain activity [20]. The collected parameters

[1] `http://vsee.com/features`

can be used to adapt the rehabilitation protocol or to remotely monitor the patient [3]. VitalJacket® (Figure 2.3) is a wearable wireless vital signs monitor. According to the user needs, it can be programmed to acquire different vital signs, such as a electrocardiogram (ECG), temperature, respiration, movement/fall and posture, among others. The collected information can be transmitted online using Bluetooth or later using a Secure Digital (SD) Card [27]. VitalJacket® provides an integrated Software Development Kit (SDK) to allow its integration into Research & Development projects.

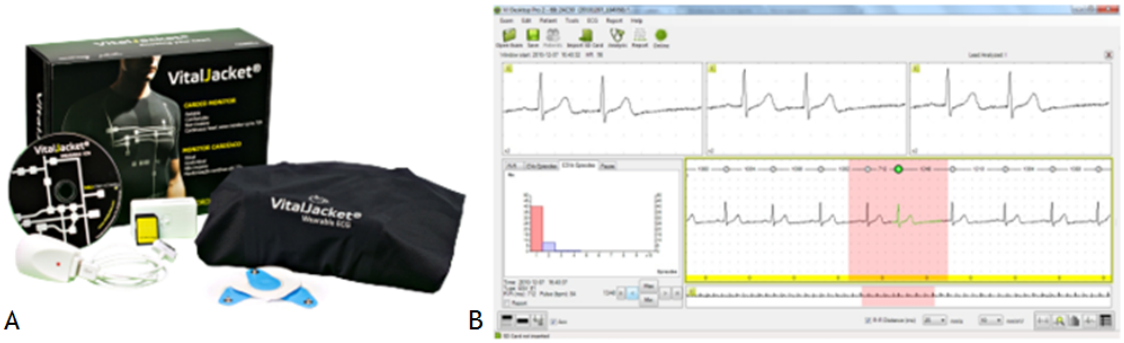

Figure 2.3: (A) Commercial package of VitalJacket®. (B) Software application to monitor ECG data[2].

Virtual reality (VR) can be perceived as a hybrid method that combines both sensor and software modules resulting in real-world environments and real-time interactions [3]. VR systems are capable of creating and controlling 3D environments that are not otherwise available in traditional methods [7]; hence this is one of the most promising applications for telerehabilitation systems [28]. Other advantages of VR systems include patient motivation, adaptability, online remote data access, economy of scale and reduced medical costs [5]. Virtual Rehab® (Figure 2.4) is a clinically validated rehabilitation system that combines videogame technology with patients' monitoring. The motion capture is accomplished by using Microsoft® Kinect and information sharing by using Microsoft® Azure, a cloud based platform in which a specialist supervises the rehabilitation sessions of patients.

2.2 Human Body Reconstruction

The creation of 3D human body models has been an active and ongoing research problem in the field of Computer Vision. The development of accurate 3D reconstruction methods has been witnessed allowing the production of accurate and realistic human models. A wide variety of applications demands 3D models of the human body such as the cinematographic industry, virtual clothing, realistic human animation and biomedical applications [29, 30]. Regarding the biomedical field, the problem of human body reconstruction has been applied, for example, in the assessment of breast cancer reconstructive surgery [31], body fat estimation [32] and evaluation of

[2] http://www.vitaljacket.com/
[3] http://www.virtualrehab.info/

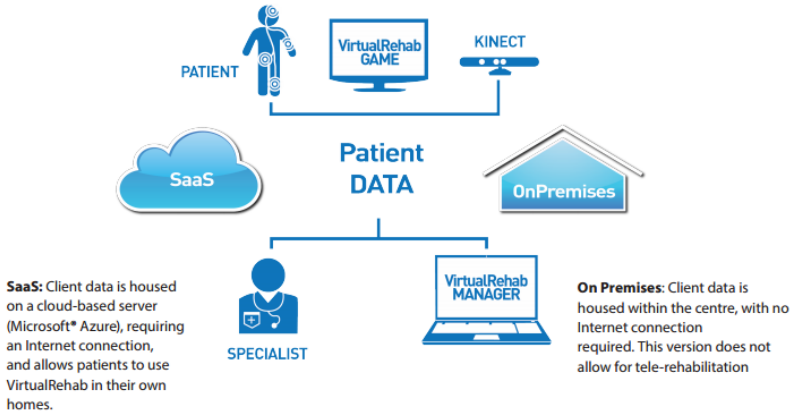

Figure 2.4: Schematic representation of the services provided by the Virtual Rehab® software[3].

maxillofacial surgery outcome [33], among others. From a telerehabilitation perspective, the reconstruction of the human body is useful since it is often used as a preprocessing stage to allow the continuous detection and tracking of human movements as described in detail in Section 2.3.

2.2.1 Depth Map Based Methods

Overall 3D reconstruction methods can be divided into contact and non-contact methods. Non-contact methods include, among others, the optical methods presented in Figure 2.5. The three examples provided were chosen since in the last decade, the wide spread of commercially available cameras boosted their use for 3D human body reconstruction. The presented optical sensors can be further divided into active and passive methods. The main difference between the two is that for active methods the illumination source, which includes some form of temporal or spatial modulation, is controlled while for the passive methods illumination is only controlled to ensure better quality. In active methods the special illumination is used to simplify the capturing process and so they tend to be less demanding regarding computational costs. On the other hand, since they require special illumination the environments in which they can be used are more restricted [34].

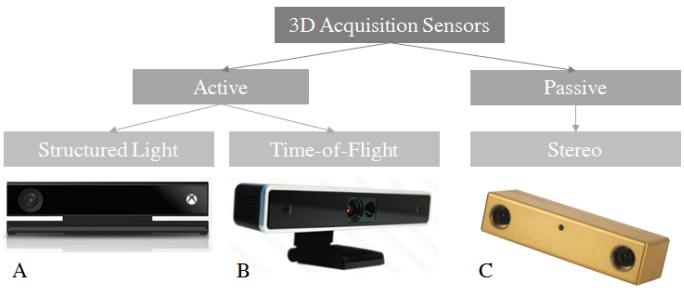

Figure 2.5: Optical methods for extracting 3D information and respectively commercially available cameras. (A) Microsoft® Kinect v2[4]. (B) SoftKinect DepthSense DS311[5]. (C) Bumblebee®2 from Point Grey Research[6]. (Adapted from [34].)

Passive stereo vision systems mimic the human vision and gather the 3D perception of a scene by triangulating corresponding points provided from two (or more) viewpoints [11]. Active methods include structured light and ToF sensors. Structured light sensors extract the 3D information in a similar way to passive stereo vision, but they replace one of the cameras by a light source that projects a known pattern into the scene [11]. ToF sensors, on the other hand, acquire depth information based on the duration that a light pulse takes to be reflected from an object [34, 11].

The mentioned acquisition sensors present some advantages and drawbacks, summarized on Table 2.1. Structured light sensors are less expensive than ToF sensors, however their depth maps present holes because some image points are not acquired by both the camera and the projector. As well, structured light sensors perform poorly when transparent, absorptive and reflective objects are present within the scene. Also, when more than one RGB-D camera is used with overlapping views, the projected patterns interfere with each other since the camera is unable to distinguish its own. This decreases the quality of the resultant depth map [17]. Despite of producing a depth map that covers every pixel, ToF sensors possess low resolution. Stereo vision cameras have the advantage of not having limited range or field of view. Also the object can be extracted within its natural environment. However, stereo systems are sensible to illumination changes and their performance is weak for non-textured and low contrast surfaces. As well, reconstructing a depth map from stereo images can be somewhat burdensome from a computational perspective [11, 35].

Table 2.1: Comparison of 3D acquisition sensors. (From [11].)

Sensor Type	**Stereo Cameras**	**Time-of-Flight**	**Structured Light**
Resolution	High: 640 - 480 or more	Low: 64 - 48 to 200 - 200	High: 640 - 480
Speed	Slow	Fast	Fast
Range	Only limited by baseline	Varies from 5 m to 10 m (indoors or outdoors)	0.8 – 3.5 m (typically indoors)
Depth Resolution	Depends on camera baseline and resolution	Less than 5 mm	Less than 1 cm
Field of View	Not limited Depends on camera lenses	Approx. 43^o (v), 69^o (h)	43^o (v), 57^o (h)
Holes in the depth map	Yes	No	Yes
Price	Cheap	Expensive	Cheap
Sensitive to Lighting	Yes	No	No

The recent developments in depth sensor hardware allowed an increase in research work that uses depth information to recover the 3D scene information and hence allow the 3D reconstruction of human figures [11]. In the past years most of the research work in human body reconstruction was based on single intensity images or image sequences. Intensity images contain rich colour

[4] `http://www.microsoft.com/en-us/kinectforwindows/`
[5] `http://www.softkinetic.com/Products/DepthSense-Cameras`
[6] `http://www.ptgrey.com/bumblebee2-firewire-stereo-vision-camera-systems`

and texture information which simplifies some image processing tasks, however as a result they are more sensitive to illumination variations. This happens because interest point detectors are more attracted to texture instead of object geometry and background subtraction can be difficult to achieve in unconstrained scenarios. In opposition, depth images are less sensible to illumination. Also, by providing a 3D calibrated structure of the scene they simplify some tasks such as background subtraction, segmentation and motion estimation [11]. Also, depth images can solve some ambiguous 2D features, like silhouette features that are used to reconstruct human body poses. As can be observed in Figure 2.6, two different body poses present similar silhouette images [36]. Nevertheless, the 3D information can be used to distinguish between the two poses.

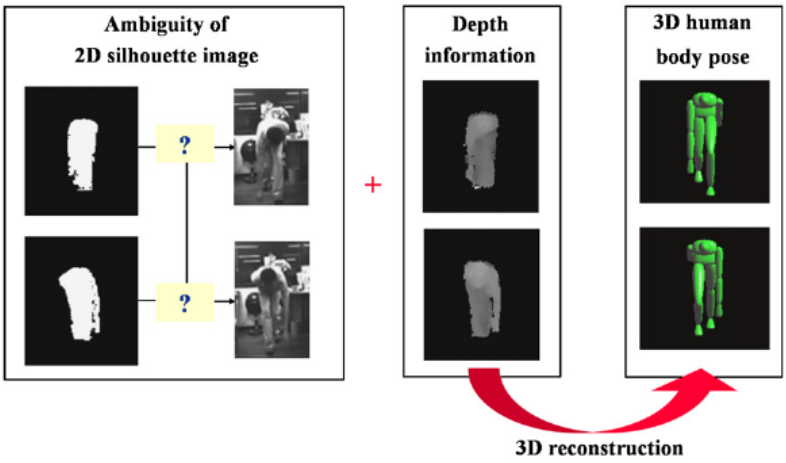

Figure 2.6: The ambiguity of 2D image silhouette features is removed by combining them with depth information to reconstruct 3D human body poses. When observing the respective depth images the ambiguity is overcome simplifying the distinction between the pose with the left leg forward from the pose with the right leg forward. (From [36].)

Like intensity images, depth images also require some pre-processing such as background subtraction, noise reduction and hole filling. By providing a 2.5 dimensional image, in which each pixel value represents a calibrated distance between the sensor and the object, background subtraction and segmentation are simplified by, for example, using distance constraints. Noise reduction can be achieved by using either morphological operations or median filtering. For example, Newcombe et al. [37] used a bilateral filter to reduce the noise and preserve the discontinuity of a Microsoft® Kinect raw depth map. In order to improve the resolution and quality of a ToF depth image, Schuon et al. [38] created LidarBoost, a super-resolution algorithm. As previously mentioned, stereo and structured light systems have the drawback of producing depth maps with points where depth is undefined (holes). To obtain more accurate and robust 3D point clouds the holes removal or reduction is required [11]; this can be achieved by estimating the depths of the depth maps holes and then filling the hole with depth-aided image inpainting based on the sparsity of the hole, such as in the method proposed by Choi et al [39]. Their algorithm also uses an edge-preserving priority to fill the patches and to allow an accurate reconstruction of background edges. The proposed method achieves better results than previous ones as can be observed in Figure 2.7.

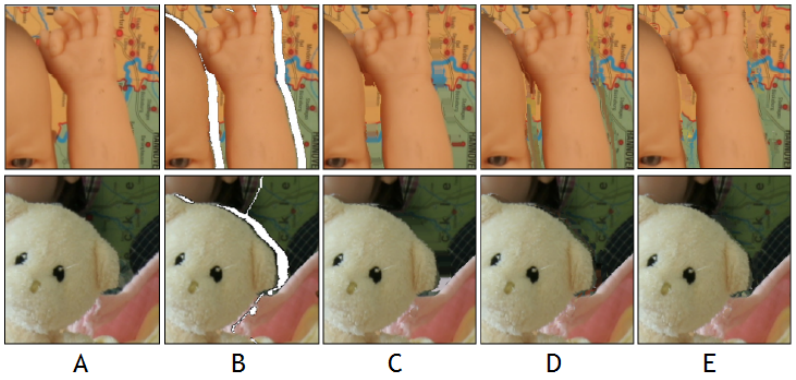

Figure 2.7: Comparison of the hole-filling algorithm results obtained by Choi et al. [39] and previous developed hole-filling methods. (A) Original Image. (B) Image with the respective hole area. (C) Po et al's algorithm [40]. (D) Gautier et al's algorithm [41]. (E) Choi et al's algorithm [39]. (Adapted from [39].)

Ziegler et al. [42] used the stereo vision Triclops SDK to acquire depth images and to obtain a point cloud that was then used for motion tracking (Figure 2.8). However, the disparity computation from stereo images in low-textured and dark image regions results in many invalid points, so the author proposed an edge detection filter to remove the invalid points. Also, foreground extraction was performed by comparing each pixel of the acquired disparity image with the corresponding pixel in a background model. If the current disparity was greater than the one in the model the pixel was considered as foreground (Figure 2.8B and C). However, after 3D reconstruction, the used foreground segmentation resulted in artefacts that were removed using a subsequent filtering stage (Figure 2.8D and E).

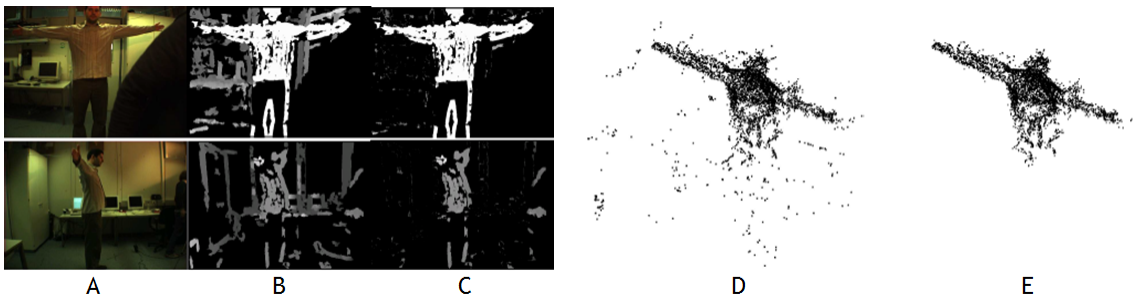

Figure 2.8: Preprocessing of stereo images proposed by Ziegler et al. [42]. (A) Original images. (B) Depth maps. (C) Result of foreground segmentation. Point cloud after (D) and before (E) noise filter application. (Adapted from [42].)

As previously referred three main sensors can be used to model 3D human bodies from depth data. The two mentioned active sensors, structured light and ToF, can be combined with the acquisition of an RGB image, being commonly known as RGB-D cameras. Microsof® Kinect is an example of a RGB-D camera based on the structured light method. Since its release in 2010, it has been widely used in 3D human body modelling, since it can acquire both depth and image

data at video speed without the need of precise lighting or texture conditions and due to its low cost, compactness and easiness to use [43].

Tong et al. [44] developed a system to scan 3D full human bodies using three Kinects and a turntable. Since one of the problems of the Microsof® Kinect is its loss of accuracy at higher distances each Kinect was used to scan a part of the human body to prevent the loss of accuracy. The total body acquisition was accomplished by positioning the person between the Kinects over a turntable (Figure 2.9A). When acquiring 3D human models the difficulty of maintaining the body still arised as another problem and so non-rigid registration was performed (Figure 2.9B).

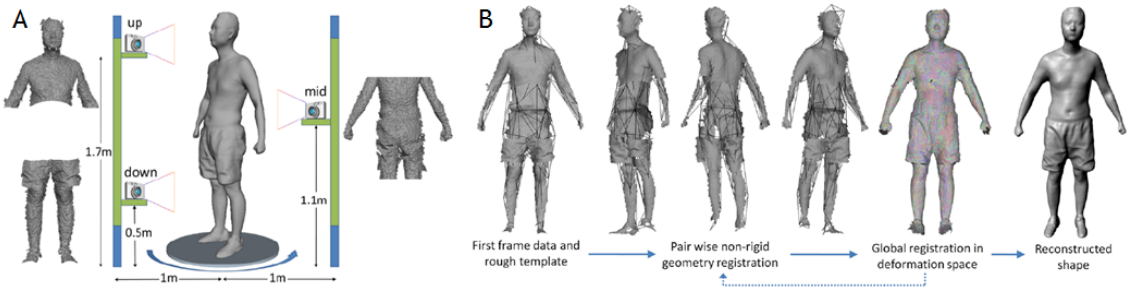

Figure 2.9: (A) Acquisition setup and (B) overview of the reconstruction algorithm of the method proposed by Tong et al. [44]. (Adapted from [44].)

Cui et al. [43] proposed a method similar to Tong et al. [44] by also using a global non-rigid registration algorithm; instead of using three Kinects, the data acquisition was performed with only one Kinect and no rough template was needed to perform the global registration. Nonetheless, the user still needed to maintain a "T" pose during acquisition and also registration was hampered by motion in the arms and legs.

When compared to other 3D acquisition methods such as structured light or stereo vision, the main advantages of ToF cameras are related to their ability to acquire depth maps in real time with small consideration for texture or lighting conditions. Since the distance is obtained by an active sensor that measures the travel time of infrared light it does not interfere with the scene in the visual spectrum. However the acquired data cannot be directly used due to the low image resolution and high noise level [45]. This situation leads to the need of an improvement of the data's quality. Many algorithms have been developed to solve this problem such as the one proposed by Bohme et al. [46]. The algorithm used a probabilistic model that improved the accuracy of range maps by imposing shading constraints and the 3D super resolution algorithm developed by Cui et al. [47].

Cho et al. [48] created a 3D human actor using a ToF camera. The authors resolved some of the main problems inherent to ToF depth data acquisition as noise reduction, recovery of lost hair region and boundary refinement. The enhanced depth images were then used to create the 3D surface of the human actor by 3D mesh triangulation. The final 3D human actor was generated by providing the surface with the corresponding color images (Figure 2.10).

On the previous mentioned example, the hair region was not directly scanned but interpolated by the boundary curves. Normally, 3D scanning methods fail to acquire the hairstyle due to its

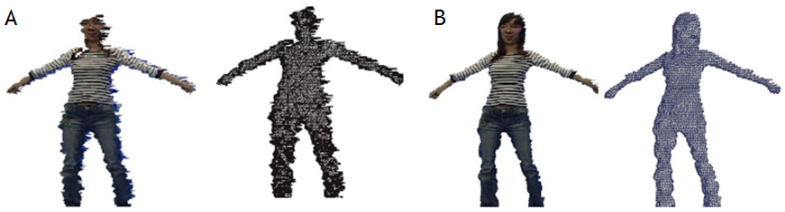

Figure 2.10: 3D human actor generated by (A) raw depth data and (B) enhanced depth data by the method proposed by Cho et al. [48]. (Adapted from [48].)

reflective properties and complicated geometry, producing models with a high level of noise. However, in the work developed by Tong et al. [45], a ToF camera was used to produce a 3D human body model with hairstyle. By using an optimization method based on the refinement of temporal average meshes followed by rigid registration the authors were able to produce accurate 3D body models with hair as presented in Figure 2.11.

Figure 2.11: Reconstructed models in different views with the corresponding images. (A) Long straight hair. (B) Long curve hair. (Adapted from [45].)

Passive stereo techniques are less used in real-time applications since one of the key stages on the stereo reconstruction pipeline, the matching between corresponding pixels in the two views, can be somewhat burdensome from a computational perspective. In comparison to the already mentioned active techniques, passive stereo vision has the advantage of not using either a laser or light pattern and so it is less sensible to sunlight interference [49].

Yu et al. [30] presented a whole body surface imaging system. In order to acquire the entire body four stereo units were used (Figure 2.12A). Each stereo unit was composed by two cameras and a projector to cast a speckle pattern to the surface body to improve texture. The authors proposed a robust sub-pixel dense stereo matching algorithm divided into two stages that was able to distinguish the body figure from its background and perform high precision matching. The obtained results proved that the algorithm was capable of producing whole body shapes with high accuracy (Figure 2.12B)

Miyazawa et al. [50] used passive stereo vision to develop a 3D scanning system. The stereo camera possessed a narrow baseline to facilitate stereo correspondence due to the small geometric

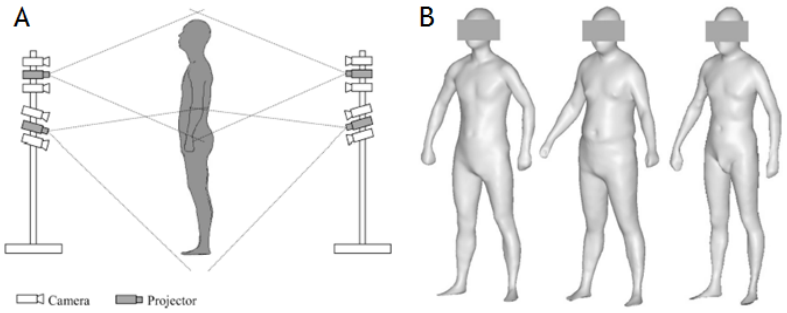

Figure 2.12: (A) Schematic representation of the stereo vision system and (B) examples of body models acquired by the system. (Adapted from [30].)

transformation between the stereo images, but as result the accuracy of the reconstructed 3D data was reduced. To overcome this problem, a high-accuracy stereo correspondence technique using phase-based image matching was applied. Also, the small baseline prevented the acquisition of the whole body from a single image and so a robot arm was used to scan the entire body (Figure 2.13A). Initially the system detected the human face and then acquired partially overlapping stereo images of the body from face to foot. After acquiring the 3D point cloud a registration process based on the Iterative Closest Point (ICP) [51] algorithm was performed. The proposed system was able to acquire high-quality 3D body information with sub-millimiter accuracy (Figure 2.13B).

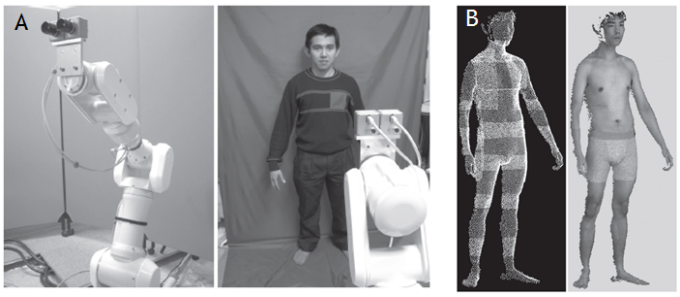

Figure 2.13: (A) 3D body scanning system proposed by Miyazawa et al. [50] and (B) captured 3D body data. (Adapted from [50].)

One of the main drawbacks of active methods is their relatively low depth resolution and high noise level which hinders the production of high-quality 3D models [11, 43]. In opposition, a calibrated stereo setup with high resolution cameras can produce depth maps of higher depth resolution [52]. Recently, hybrid methods that combine passive stereo with active methods, are beginning to arise allowing the production of high resolution depth maps.

Passive stereo systems perform badly in textureless surfaces, repeated patterns and occluded areas, while ToF cameras have some difficulties in dealing with rich texture regions (where passive stereo outperforms), but deal well with textureless surfaces. Several approaches have been

developed to merge ToF sensor data with images captured from monochromatic or stereo cameras. A recent overview of fusion methods can be consulted in [53]. Zhu et al. [54] explored the mentioned complementary nature by fusing the probability distribution function of the depth data from each sensor using a Markov Random Field [55], producing a combined sensor with higher features. The results revealed an overall error reduction of 50% when compared to state-of-the-art methods such as structured light. However the fusion approach needed close to 20 seconds to create a depth map of 400 x 300 resolution making them unsuitable for real time applications.

Similarly, Somanath et al. [52] proposed a stereo algorithm that combined the information from stereo RGB images with the low resolution depth information of Microsoft® Kinect in order to obtain a dense depth map with higher depth and spatial resolution. The depth estimates from Microsoft® Kinect and the confidences from both sensors were used to calculate the data cost only for a sparse set of labels. The matching cost also took into consideration the image consistency in combination with the geometry prior provided by the Microsoft® Kinect. Moreover, for dealing with textured and non textured surfaces, a smoothness prior was obtained trough the combination of the depth and image gradient information. The results revealed an improvement in depth resolution, recovery of small geometric details and correct depth in ambiguous areas, thus overcoming the individual flaws of the sensors.

Nevertheless, the previous mentioned examples were not specifically applied in the reconstruction of 3D human bodies, but to the reconstruction of objects. Jia et al. [56], combined the high quality of stereo image with the very fast, but low resolution, depth acquisition of Microsoft® Kinect to create an efficient and enhanced 3D image reconstruction system. The developed system was able to create real-time high resolution 3D image with 30 fps (Frames per Second) (Figure 2.14)

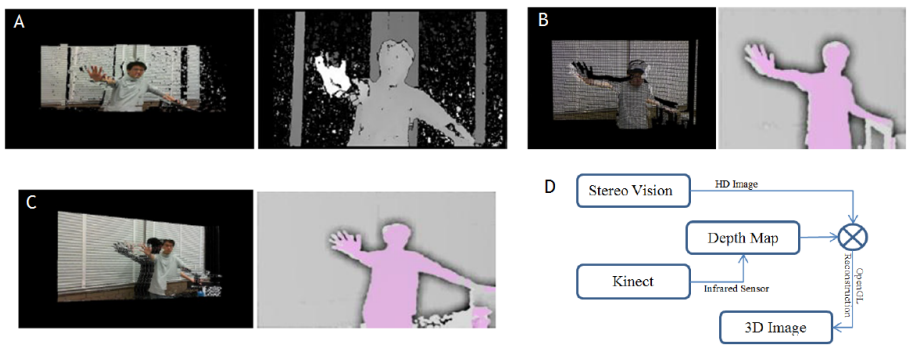

Figure 2.14: 3D reconstructed image and depth map obtained by (A) stereo vision, (B) Microsoft® Kinect and (C) combined method. (D) Design flow of the 3D image reconstruction method proposed by Jia et al. [56]. (Adapted from [56].)

2.3 Human Pose Estimation and Motion Tracking

In telerehabilitation, human motion tracking systems should be able to generate real-time data to dynamically represent the position changes of a human body (or portion of it). The tracking system can be non-visual or visual based, with markers or marker free, depending if indicators need to be connected to body parts.

Non-visual tracking systems rely on the use of sensors that are connected to the human body and collect movement information. A wide variety of sensors is available such as mechanical, inertial, acoustic, radio, microwave and magnetic sensors. Each category as its own limitations and advantages considering the environment in which they are used. The main advantage of non-visual systems is that they do not suffer from the "line-of-sight" problem, as the visual systems do, where an unobstructed view must be maintained between the robot and the camera [6].

Visual tracking systems take advantage of optical sensors (e.g. cameras) to improve the accuracy in pose estimation and can be divided into marker and marker free systems. Visual marker based systems (MBS) follow the human movement by using cameras and identifiers (markers) located in the human body surface. During body movement, since the human skeleton is a very complex structure each body part performs its own motion trajectory with high degrees of freedom. This complexity proves to be a drawback in consistent and reliable motion estimation. The use of markers solves this problem by minimizing the ambiguity in the subject's movements. However, these systems are expensive and, in a home setting, attaching the markers or dressing a suit can become a burdensome task [57]. Other problems arise from the use of markers like the addition of noisy data due to the movement of skin under the markers or movement of the marker itself resulting in unreliable landmarks [6]. Also, image acquisition when using MBS is often limited to a specific laboratory setting [35] which constitutes a major drawback in a telerehabilitation application. As well, one of the main problems of MBS is *reproducibility* due to the variation of marker placement between sessions [9]. Markerless systems (MLS) can overcome most of these problems since they only rely on the information given by the visual sensors and do result in a less restrictive system. Cameras can deliver high accuracy in movement tracking due to their high resolution and their parameters can be easily estimated. Also, nowadays they are relatively inexpensive, can be modular and adaptable to different scenarios and mainly are non-obtrusive, which is a very important feature in a telerehabilitation setup [6]. However, MLS are more sensible to changes in illumination and appearance [57]. MBS are still the gold standard for motion tracking since they can deliver an higher precision and are often used on the evaluation of MLS [9, 58, 10].

MLS can be further divided into two approaches: model based and model free which are further detailed in the next sections. Also, when considering a motion capture system several subprocesses can be included such as initialization, data pre-processing, tracking and pose estimation [59]. Initialization includes the preparation for the actual motion capture system and for example, the synthesis of the articulated models in the model based approach is part of this process. Data pre-processing gathers the information needed in order to perform the pose estimation and includes the segmentation of the human body within the image sequences and the 3D reconstruction, when

necessary. Finally, pose estimation uses the processed information to acquire the pose parameters [35].

2.3.1 Model Based Approaches

Model based approaches are the most common within motion capture systems. This approach fits a 3D representation of the human body, which includes a surface point cloud and an associated skeleton representation, to the acquired data, that can be either 2D or 3D information. The joint centers and their velocities can then be estimated by using the articulated skeletons. Also, computation time and algorithm complexity may be reduced by implementing constraints to the joints like boundaries to the range of movement due to dependency of neighbouring joints [35].

Most of the model based approaches require an initialization step in which the model used to fit the 2D or 3D acquired data is constructed. The simpler the model, more computationally fast and simple it is to implement, but it is more prone to errors and deviations [35]. Kohli et al. [60] used a primitive model for segmentation and pose estimation of human figures. The presented model included a rectangular torso with sticks as limbs (Figure 2.15A1), 26 degrees of freedom (DoF), but no constraints. In order to obtain a probability distribution of whether the pixels belonged to the silhouette, the simpler model was replaced by a prior shape (Figure 2.15A2). The segmentation results were promising, but the accuracy in the pose estimation was not strong enough according to biomechanical needs.

Ogawara et al. [61] proposed a more advanced deformable articulated model and fitted him to a 3D reconstructed volume obtained from multiple video streams (Figure 2.15B). The used model was composed by a link model for joint structure representation and a skin model for body surface representation (naturally deformed as a consequence of changes in the joint angles and so constrained the deformation and prevented unnatural movements). The deformable model had 29 DoF for joints, 3 DoF for body translation and 3 DoF for body rotation.

Figure 2.15: (A) Primitive models proposed by Kohli et al. [60]. (A1) Stickman and (A2) corresponding prior shape (distance transform). (B) Surface model proposed by Ogawara et al. [61]. (Adapted from [60, 61].)

Human shapes can also be synthesised using repositories such as in the Shape Completion and Animation of PEople (SCAPE) method proposed by Anguelov et al. [62]. This method included both articulated and non-rigid deformations and built two separate models that could be combined

to produce 3D surface models. One model derived the variation in body poses while the other derived the variation in body shapes. The variation in pose was separated into a rigid and non-rigid component and the deformation was constrained by the movement of adjacent joints which significantly reduced the dimensionality of the model. Principal component analysis (PCA) [63] was used to represent the shape variation by inducing a low-dimensional subspace of body shape deformations. The dataset is available online[7] and has been used in a wide variety of human shape and pose estimation tasks [64, 65, 66, 67, 68]. However, SCAPE has the drawback of requiring a high-quality initialization shape (either from marker-based motion capture or Visual Hull (VH) from a surrounding camera array) and the outcome quality is highly dependent on the quality of the training data [69].

When the use of repositories is unsuitable, subject-specific models can also be automatically generated such as in the system proposed by Corazza et al. [70] (Figure 2.16). The subject-specific model was created based on an automatic model generation algorithm [71] and a SCAPE model with a biomechanically consistent kinematic model and a pose-shape matching algorithm, being generated from a single static recording. The obtained accuracy when comparing to anatomical landmarks was under 2.5 cm.

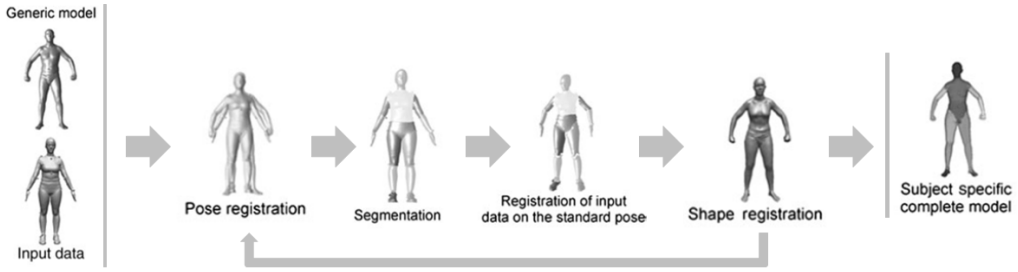

Figure 2.16: Overview of the iterative pose registration method proposed by Corazza et al. [70]. (Adapted from [35].)

Many of the pose estimation methods use a non-rigid registration approach to fit a body or skeleton model to 3D data. For tracking body motion from 3D mesh fitting, ICP is a popular method [11]. The ICP was originally developed by Besl et al. [51] to align rigid 3D shapes. Normally, by using a nearest neighbour principle the point correspondences are found between the articulated object and the reference model and are then used in a least minimization problem.

Grest et al. [72] estimated motion from a point cloud obtained from depth data acquired from stereo images, Figure 2.17. The body pose estimation was accomplished by using an analytically derived Jacobian and the correspondences between the point cloud and the model points were obtained using ICP. The proposed method achieved real time performance.

Other methods propose ICP based approaches with some modifications such as the one proposed by Chen et al. [73]. The authors used a Hierarchical Weighted Iterative Closest Point (HWICP) method to fit an articulated model to a Visual Hull model (Figure 2.18). According to the authors when the adjacent body segments are near cylindrical-shape the general hierarchical

[7] `http://ai.stanford.edu/~drago/Projects/scape/scape.html`

Figure 2.17: Overview of the method proposed by Grest et al. [72]. (A) Body model (left) and joints of the arm (right). (B) Depth images (right), original image overlayed with the estimated body pose (center) and model view from one side (left). (Adapted from [72].)

ICP fails [71]. For this reason, they suggested HWICP in which the voxels of the skin model were weighted and only the heavily weighted were included for ICP registration. Regions prone to noise were assigned low values whereas regions with rapidly changing surface normals were assigned higher weights. The results presented were promising and the authors were able to estimate the movements of limbs with 6 DoF for all joints [73].

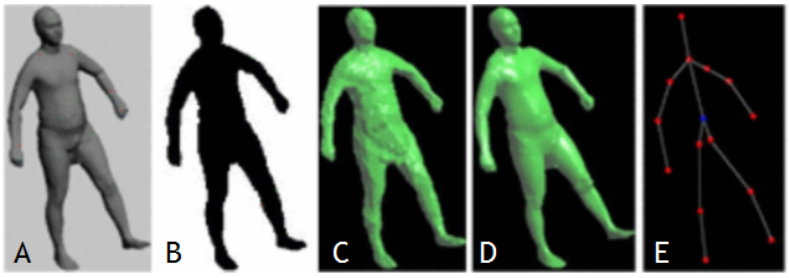

Figure 2.18: Overview of the method proposed by Chen et al. [73]. (A) Target, (B) Silhouette, (C) Visual Hull, (D) 3D human body shape tracking, (E) Motion tracking. (From [73].)

Despite the popularity of ICP, it has the major drawbacks of requiring a good initial position and not being able to handle tracking failure [11]. Other approaches make use of alternative registration procedures. Ye et al. [69] estimated pose from a single depth image and traded real-time performance for accuracy. In the proposed method, a motion database, created from a generic human mesh model, was used. The human model included not only the surface mesh but also the corresponding skeleton containing a total of 19 joints. An overview of the reported solution can be observed in Figure 2.19. Having as input a point cloud (or a depth map) the object of interest was segmented using distance constraints and then the noise was removed by applying a modified surface reconstruction algorithm. Next, the Coherent Point Drift (CPD) [74] algorithm was used to estimate a refined pose configuration. The occlusion problem was solved by using the database information. The final output was accomplished by a failure detection and recovery mechanism using temporal information. Results revealed an accuracy of 38 mm in joint detection.

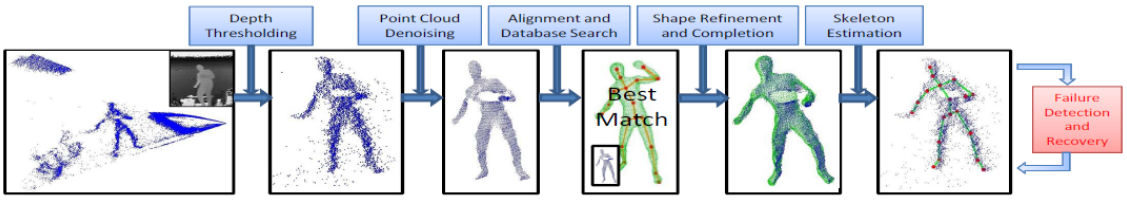

Figure 2.19: Overview of the method proposed by Ye et al. [69]. The result is an estimated skeleton embedded in the input point cloud. (From [69].)

Schwarz et al. [75] tracked full body movements from depth images using geodesic distances and optical flow. First, anatomical landmarks were identified in the depth data and were then used to fit a skeleton using inverse kinematics. The body part detection was facilitated by representing the 3D points on the surface of a person as a graph in order to measure the geodesic distances between different points in the body. Geodesic distances were chosen instead of euclidean distances since they are independent of body posture. The occlusion problem was solved using an optical flow approach computed on the depth images. The results were evaluated using both input data provided by Kinect and a ToF camera and reported that the proposed method was able to estimate 3D full body poses with high accuracy.

More recently, particle filters are being used to solve the problem of motion capture. Particle filters allow the estimation of bayesian models that consist in a set of latent variables connected in a Markov chain. The latent variables are estimated by simulation using an arbitrary number of particles. In a motion capture system, the latent variables are the pose parameters and images in the video sequence and the 3D reconstructions are the observed data. An efficient distribution of the particles is accomplished by using the solution from the previous frame as prior information [35].

Li et al. [76] proposed a motion capture system that, by having a stereo input, was able to track human motion with a particle filter approach with partitioned sampling in order to solve the high dimensionality problem. A quantitative error analysis was accomplished by using videos from the publicly available CMU MOCAP database[8] that includes ground truth data obtained using a marker-based motion capture system. Results revealed a good accuracy even with random, fast and complex motions at a near real time speed.

Despite the advances achieved regarding particle filter approaches, most of them suffer from the curse of dimensionality and need to use simple human models (that result in suboptimal tracking outcomes) or need a high number of evaluations to achieve accurate results. The latter problem is of key importance in real-time applications since in order to obtain an adequate speed the optimal result should be found in as few iterations as possible. Based on a stochastic optimization approach, evolutionary algorithms are appearing as an option to particle filter approaches. Evolutionary algorithms present some advantages such as a robust and reliable performance, global and local search capability, little or no information requirement [77]. Bolivar et al. [77] compared

[8] `http://mocap.cs.cmu.edu/`

the performance of three relevant evolutionary algorithms namely Covariance Matrix Adaptation Evolutionary Strategy [78], Differential Evolution [79] and Particle Swarm Optimization (PSO) [80], with two commonly used variants of particle filters, Annealed Particle Filter [81] and Partitioned Sampling Annealed Particle Filter [82]. The results demonstrated that generic optimization algorithms provided significantly better results than particle filter approaches.

Michel et al. [83] considered human body tracking as an optimization problem and implemented the PSO method, Figure 2.20. The problem was solved by using a top-down approach that minimized the discrepancy between the 3D occupancy of hypothesized instances of a human body model and the volume reconstructed from the observations. The method input was a volumetric representation of the human body acquired by the fusion of the information provided by the depth map of two Microsoft® Kinects. As stated by the author the input data could also be a volumetric representation obtained by computing the VH. The authors proposed three variants of the PSO method, baseline PSO (bPSO), perturbed PSO (pPSO) and HYBRID, that combined the pPSO with the OpenNI method [9] and did not required an initial body pose and knowledge of the body model parameters. The results revealed that the pPSO outperformed the bPSO method being more accurate than the OpenNI. Regarding accuracy, the HYBRID approach was slightly less accurate. However, due to the absence of the need of a initialization process, the HYBRID method was considered more practical.

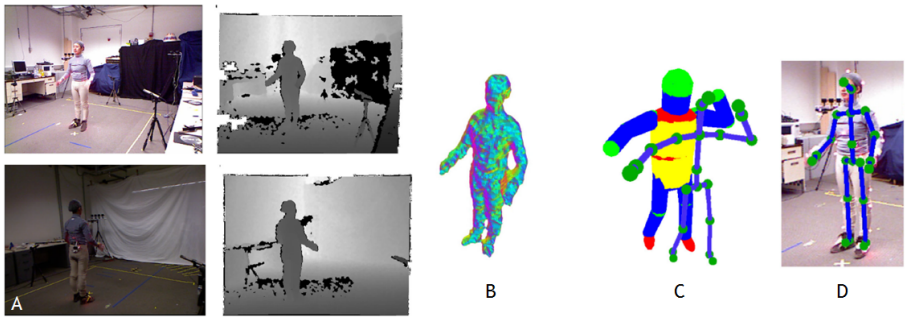

Figure 2.20: Overview of the method proposed by Michel et al. [83]. (A) Two RGB frames (left) and respective depth maps (right). The volume (B) occupied by the person is obtained using the depth maps. Then the proposed method fits the used human body model (C) to this volume, recovering the body skeleton (D). (Adapted from [83].)

2.3.2 Model Free Approaches

Through the use of supervised learning, model free approaches relate the observed data to the pose by searching for an image that is the best match to an input image from the training set [36, 35]. One of the main drawbacks of these methods is the fact that if a pose is not included within the training set, they will most likely fail to estimate it [35].

Yang et al. [36] implemented a learning based approach to track human body poses from stereo images, Figure 2.21. The authors used a linear combination of prototypes of 2D depth images and

[9] http://wiki.ros.org/openni_tracker

their corresponding 3D joint positions to represent the human body pose. In order to reduce complexity, during the learning stage the human body poses were hierarchically divided into several sub-clusters. Depth images were used to overcome ambiguities and a top-down learning approach allowed the reconstruction of 3D human body poses that were not present in the training data. However, the proposed method was unable to deal with other viewpoints rather than the frontal view.

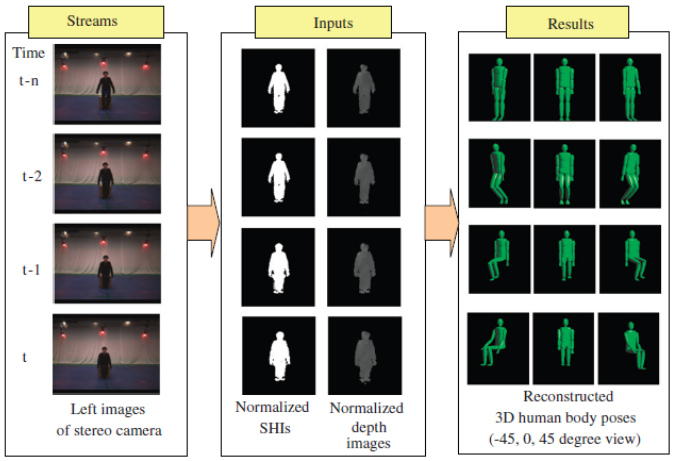

Figure 2.21: Examples of the tracked 3D human body poses with a sitting on a chair sequence using the method proposed by Yang et al. [36]. (From [36].)

Shotton et al. [84] developed a new human pose recognition method from a single depth image divided into two stages: body part labelling and 3D joint position estimation, Figure 2.22. First a dense probabilistic body part labelling, whose parts were spatially localized near skeletal points of interest, was done by using a segmented depth image. The labelling was accomplished using per pixel classification based on Randomized Decision Forests (RDF) [85] trained using a large database of synthetic depth images. This approach allowed the identification of up to 31 body parts in real-time. Then, a mean-shift algorithm was used to find the spatial modes of each part distribution resulting in confidence-weighted proposals for the 3D locations of each skeletal joint. The body part recognition algorithm described in [84] comprises the initial part of the method used by the skeletal tracker provided with the Microsoft® Kinect. The second part, in which a skeleton is fitted to the hypothesized joint positions is part of an unpublished proprietary algorithm. Kinematic and temporal constraints are explored in order to obtain a smoothed output skeleton that is able to handle occlusions [86].

Estimating the joint position using the mean-shift algorithm has some drawbacks: the size and shape of the subject deeply influences the joint position estimation, the relative information obtained is related to the body surface, whereas joints are localized inside body parts [87]. In order to surpass these limitations Dinh et al. [87] developed a new 3D body pose recovery approach based on Principal Direction Analysis (PDA) of recognized human body parts from a series of depth images, Figure 2.23. First, trained RDF were used to identify the human body parts within a synthetic

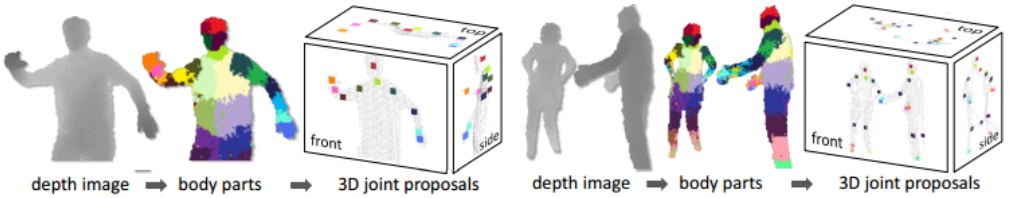

Figure 2.22: Overview of the method proposed by Shotton et al. [84]. Using a single depth image a per-pixel body distribution is inferred. Then, high-quality 3D proposals for the locations of each body joint are obtained by estimating local modes. Both single an multiple detection is possible. (From [84].)

training database. The recognized body parts were used to estimate the principal direction vectors using PDA. Finally, the 3D human body pose was recovered by mapping the directional vectors to each body part of the 3D human body model. In comparison with the method proposed by Shotton et al. [84], instead of using a human skeleton model without constraints, the author proposed a more advanced model that used a kinematic chain with predefined DoFs for each joint. The more complex model represented with more feasibility the body movement and obtained more robust results, Figure 2.24. Overall results revealed that the proposed method was able to deal with sequences of unconstrained movements of persons with different sizes and shapes. Similarly to the previous mentioned approach other 3D body pose recovery methods use a learning methodology to recognize each body part [88, 89].

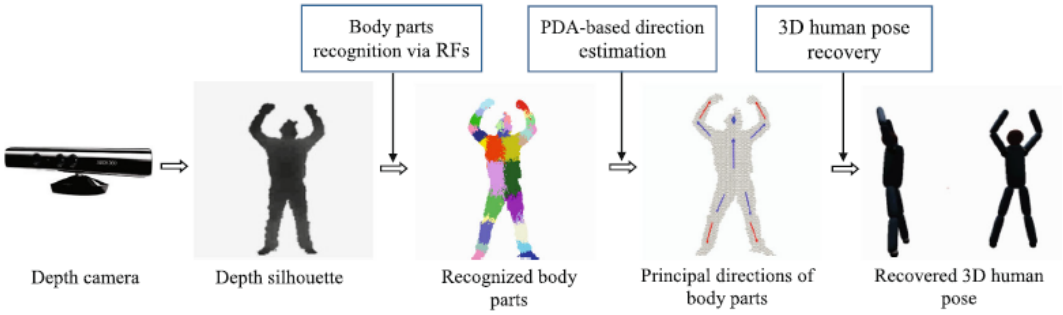

Figure 2.23: Processing pipeline of the method proposed by Dinh et al. [87]. Using as input a depth image without background the body parts are labelled and by applying PDA to the body parts the final 3D human body pose is recovered. (From [87].)

More recently, the developers of the Microsoft® Kinect skeletal tracker proposed two enhanced algorithms [86]. Girshick et al. [90] developed an algorithm that performed the regression directly on the raw depth information, instead of on the body part labelled intermediate stage. The algorithm was able to estimate the position of occluded joints. Results revealed that the implemented algorithm outperforms state-of-the-art implementations, such as the one of Shotton et al. [84], and was able to run at a speed of about 200 frames per second [90]. Taylor et al. [91] extended the initial machine learning approach by estimating correspondences directly between images pixels and a 3D mesh model, Figure 2.25. This was accomplished by employing a re-

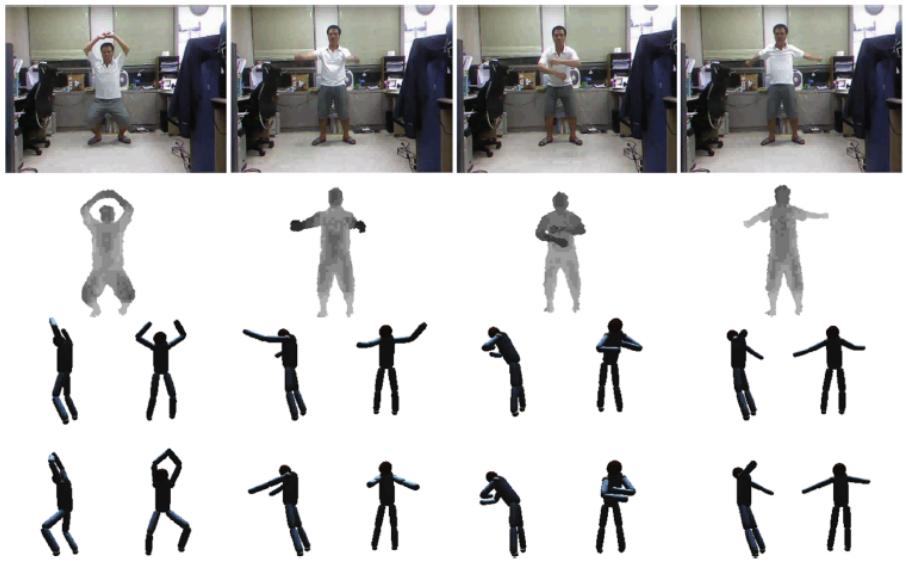

Figure 2.24: Comparison of the approach proposed by Dinh et al. [87] with the one proposed by Shotton et al. [84] for four different poses. The first row shows RGB images, the second row shows depth silhouettes, the third row shows the results obtained from the mean shift algorithm (Shotton et al. [84]) and the fourth row shows the results obtained using the PDA algorithm (Dinh et al. [87]). (From [87].)

gression forest in an energy minimization approach. Unlike ICP methods, the proposed one does not required more than one iteration for optimization since the regression forest was able to accurately estimate correspondences, thus enabling a "single-shot" optimization [91]. As an additional contribution the authors proposed a more realist evaluation metric, instead of the mean average precision used by [84] and [90]. The developed algorithm achieved a 45% score in opposition to the 20% score obtained by previous state-of-the-art algorithms [84, 90].

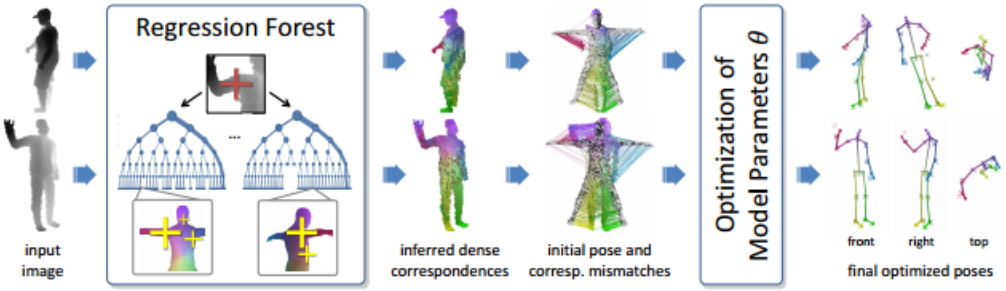

Figure 2.25: Overview of the method proposed by Taylor et al. [91]. The correspondences are estimated directly between images pixels and a 3D mesh model. Without separate initialization or alternating minimization of pose and correspondence, a fast and reliable convergence to a good pose estimate can be obtained in a "single-shot". (From [91].)

Shen et al. [92] tried to overcome the difficulty of the method of Shotton et al. [84] when dealing with severe occlusions. By using the output of the Microsoft® Kinect system as an initial estimation of the human pose, the authors improved the initial pose by exploring temporal motion

consistency and systematic bias. Ground truth poses were obtained from a mocap system[10] and combined with the direct output from the Microsoft® Kinect to train a large dataset of specific human actions. The training information was then used to correct the initial Microsoft® Kinect poses by using a random forest regressor [92].

Building on the work of Shen et al. [92], Zhou et al. [93] surpassed the need of a large training dataset. The authors used a probabilistic model based on Gaussian Process (GP) [94] to reconstruct poses directly captured by the Microsoft® Kinect system. The use of a GP based model allowed the use of a smaller training set. Results revealed that the system was able to deal with severe self-occlusion situations and outperformed the one proposed by Shen et al. [92].

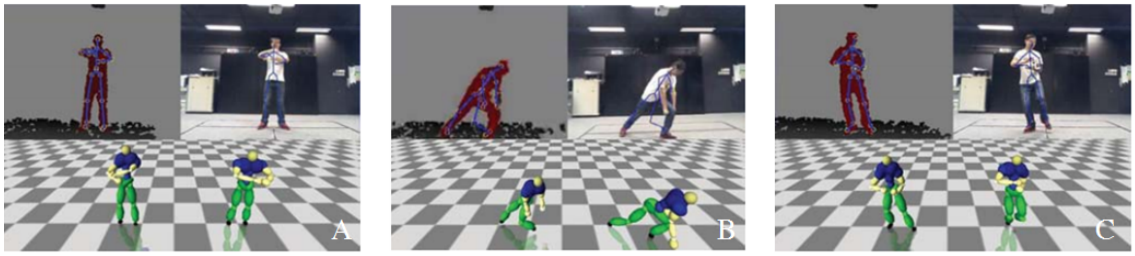

Figure 2.26: Postures from Microsoft® Kinect (left avatar) and their corresponding reconstructed poses (right avatar). The skeleton data presented in blue in the top two pictures is the tracked Microsoft® Kinect results. (A) Rolling hands forward and backward. (B) Bending leg. (C) Taichi motion. (Adapted from [93].)

2.4 Tools and Software Libraries

Recently, the appearance of low cost 3D sensing hardware boosted the interest in solving Computer Vision problems related to 3D processing. For this reason, to allow the advance in vision research and the dissemination of vision knowledge, it is of outermost importance the development of libraries of programming functions. This libraries should contain optimized and portable code and hopefully be free [95]. By gathering functions that solve basic problems, the development of open-source libraries allows the researchers to focus on solving and improving on-going problems. Some of the most commonly used libraries for Computer Vision and 3D processing handling are described in the following sections.

2.4.1 OpenCV

Open Source Computer Vision Library (OpenCV)[11] is an open source library for image and video processing that was originally introduced 15 years ago by Intel. Nowadays it is one of the most commonly used open source libraries with more than 2.5M downloads [95]. OpenCV can be used in both academic and commercial applications under the Berkeley Software Distribution

[10] http://www.vicon.com/
[11] http://opencv.org/

(BSD) license. It can be integrated with several operating systems such as Windows, Linux, Android/iOS and has C++, C, Python and Java interfaces [95]. Several state-of-the-art algorithms for image and video processing are incorporated and ready to use in OpenCV, such as algorithms for egomotion estimation, gesture recognition, segmentation, motion tracking, stereo vision and object identification, just to name a few. A detailed description of many of the algorithms and methods implemented in OpenCV can be found in [96]. The most recent stable version,released in June 2015, is 3.0.

2.4.2 Point Cloud Library

Point Cloud Library (PCL)[12] is a C++ open source library intended for 3D point cloud processing. It can be used under the BSD license, is fully integrated with the Robot Operating System (ROS) and can be ported to Windows, Linux, MacOS and Android/iOS. Among others, PCL provides state-of-the-art algorithms for filtering, feature estimation, surface reconstruction, model fitting, segmentation and registration of 3D information. The most recent stable version is 1.7.2 and was released in September 2014.

This library relies in three main third party dependencies. All the k-nearest neighbour operations are based in the Fast Library for Approximate Nearest Neighbours (FLANN)[13]. The information is passed between modules and algorithms using Boost[14] pointers. Visualization of n-D point cloud structures is accomplished by using the VTK[15] library. Also, other three libraries (Eigen, Qhull and OpenNI) can be compiled together to allow the performance of several secondary algorithms. Operations related with linear algebra, matrix and vectors are solved using Eigen[16]. The Qhull[17] library is used for convex hull and Delaunay triangulation, among others. The interface with PrimeSense devices is done through the OpenNI[18] [97].

In order to simplify the development, PCL is separated in smaller libraries that can be built separately (Figure 2.27).

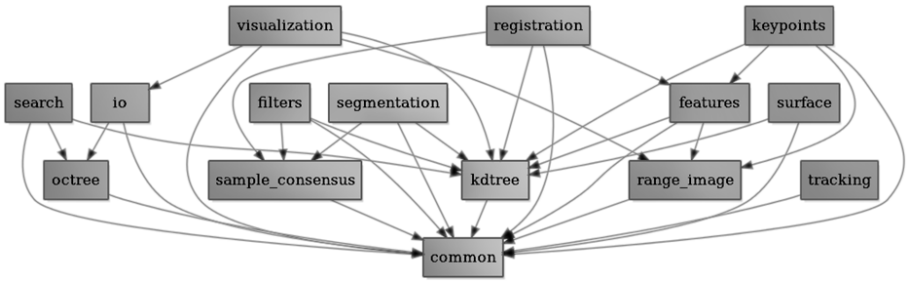

Figure 2.27: Sub-libraries of PCL. (From [98].)

[12] http://pointclouds.org/
[13] http://www.cs.ubc.ca/research/flann/
[14] http://www.boost.org/
[15] http://www.vtk.org/
[16] eigen.tuxfamily.org/
[17] http://www.qhull.org/
[18] http://structure.io/openni

2.4.3 Skeleton Tracking Libraries

With the development of affordable cameras that are able to retrieve the depth information from the viewing scene, the task of motion capture has been eased. As a consequence the creation of open source software libraries that can perform skeleton tracking (ST) has also been promoted. Table 2.2 presents some of the available libraries with their main advantages and disadvantages.

Table 2.2: Comparison between ST libraries. (Adapted from [99].)

ST Libraries	Pros	Cons
Microsoft Kinect SDK	1. Easy to install, fairly widespread. 2. Supports skeleton tracking (20 joints). 3. Does not require camera calibration. 4. Predictive tracking of joints. 5. Fast skeleton recognition. 6. Joints occlusion handled.	1. Support for Windows only. 2. Only for C/C++ and C#. 3. Higher processing power.
OpenNI	1. Support skeleton tracking (15 joints). 2. Available for most languages. 3. Any OS compatible.	1. Calibration pose required. 2. No predictive tracking. 3. Joints occlusion not handled properly. 4. Gets confused with very fast movements.
Libfreenect	1. Any OS compatible. 2. Available for most languages.	1. No skeleton tracking. 2. Difficult to install.
CL NUI	1. Can capture a wide range of body movements. 2. Camera noise can be filtered.	1. Cannot perform motion prediction. 2. No support for occlusion handling.
Evoluce SDK	1. Support various gesture recognition methods. 2. Support skeleton tracking.	1. Only for Windows 7. 2. Calibration pose is required. 3. Only for C/C++ and C#.
Delicode NImate	1. Quite fast. 2. Support skeleton tracking. 3. Does not require camera calibration.	1. Skeleton tracking not done properly. 2. Only for Windows.
GPU People (PCL)	1. Any OS compatible. 2. Open architecture. 3. Device Independent.	1. Only for C++. 2. Training data only available for Microsoft® Kinect.
Skeltrack	1. Open architecture. 2. Device Independent.	1. Only for C. 2. Can only track the upper body (7 joints). 3. Skeleton tracking not done properly.

From the presented libraries, the most robust and widely spread are the Microsoft® Kinect SDK and the OpenNI. Microsoft® Kinect SDK was released by Microsoft and its current version, compliant with Microsoft® Kinect v1, is 1.8. OpenNI works with a compliant middleware called NITE and its highest version is 2.0. OpenNI's skeletal tracker requires the user to perform a predefined calibration pose until enough joints are recognized. The time needed to hold the mentioned pose depends on the scene and processing power. Microsoft® Kinect SDK, on the other hand, does not require an initial calibration pose. For this reason it is more prone to misidentify environment objects as skeletal joints when the pose is too complicated or the environment is too

cluttered. The most recent version of Microsoft® Kinect SDK presents the option to track only the upper body of the subject. This can be advantageous for tracking people in seating position, such as patients in a wheelchair. The OpenNI's skeletal tracker is able to recognize 15 joints while the Microsoft® Kinect's can retrieve 20 joints (or 10 for the upper body tracking option), as presented in Figure 2.28. Also, it performs simple gesture recognition while OpenNI is more focused on hand detection and hand-skeletal tracking [86].

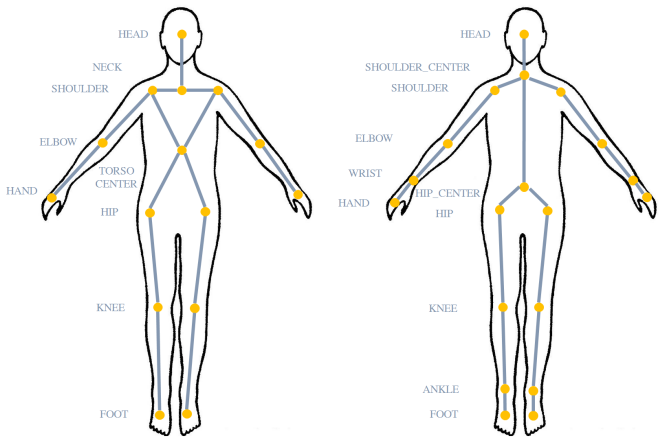

Figure 2.28: Skeleton models. (Adapted from [69] and [9], respectively.)

As can be perceived by Table 2.2, most of the available libraries rely on the information provided by a structured light based device from PrimeSense, such as the Microsoft® Kinect. As well most of the developed algorithms are *closed*, meaning that we can only access the methods to invoke then, but cannot perform any direct improvements. Nevertheless, most recently some libraries that are device agnostic have appeared. This is the case of the PCL GPU People module released by Point Cloud Library in September 2014. This module is prepared to receive the information in Point Cloud Data (PCD) format. However it relies on training data that, as of the time of writing, is provided only for Microsoft® Kinect like cameras. Skeltrack is another recent library that is device independent. However it is only capable of tracking the upper body (7 joints) and it is still in a very premature stage of development [100].

2.5 Final Discussion

Recently, 3D acquisition systems are gaining undeniable importance in 3D human body reconstruction, due to their affordability, efficiency, compactness and easiness to use. Nevertheless, they still present some drawbacks. Active sensors are more efficient in providing depth information and so more oriented for real time applications. However, they produce low resolution depth maps which is a major disadvantage for applications that require a high quality depth map. On the other hand, passive stereo vision results in higher quality depth images, but can be inefficient

from a computational perspective. Yet, recent research projects are working in the development of stereo algorithms that can achieve real time speed. Also, passive stereo has difficulties in dealing with untextured surfaces which present no difficulty for active sensors. On the other hand, active sensors have problems with very textured surfaces were passive stereo outperforms. Some hybrid methods take this situations into consideration and combine active with passive methods in order to overcome their individual weaknesses.

When considering the task of human pose estimation and motion tracking, two paths can be chosen: marker based or markerless approaches. While the first yields more accurate results, the task of attaching markers to the body normally needs to be performed by a specialized technician and also the image acquisition is often limited to a laboratory setting. These drawbacks make marker based systems unsuited for telerehabilitation applications. Markerless systems, on the other hand, are less restrictive and can be easily adapted to the home setup. However, they are more sensible to illumination and appearance modifications. Further, markerless systems can be divided into model based and model free methods. Model free approaches do not require *a priori* knowledge, but are restricted to estimating poses that are very close related to the ones presented in the training dataset. To overcome this limitation, an expansion of the dataset can allow a more flexible range of motions but at the expense of computational power. Both model free and model based approaches present the limitation of being compromised when the observed subject wears baggy clothes leading to an unprecise joint location determination. Also, most of the body models and datasets used are from healthy adults. In a telerehabilitation setup the observed subjects may present health related body deformities that would hinder the process of pose estimation. For this reason, creating a set the most varied possible regarding body composition, as to reflect the population variation, is of key importance. Considering accuracy both methods provide similar results. Nevertheless, regarding processing time, model free approaches produce the best results. When the goal is to develop a telerehabilitation application both accuracy and processing time should be considered. A good accuracy is of key importance to obtain clinical relevant information and an adequate processing time is fundamental to provide information in real-time. For this reason an adequate tradeoff between accuracy and processing time should be guaranteed.

In summary, there is no overall solution that is able to tackle all the aforementioned problems. Each application has its objectives and requirements. The developer needs to chose and improve the algorithms and methods that best suit the final system requirements, such as the texture characteristics of the scene, the need for high quality depth maps, the desired level of fine details, the accuracy and time cost efficiency in pose estimation.

Chapter 3

Methodology

One of the main goals of the present thesis was the development of a system able to track the skeleton movements of a patient in the context of a telerehabilitation approach. As described in Figure 3.1 the developed system comprised two hierarchical stages. First, a stereo camera was used to allow the acquisition of a 3D representation of the human body. Then, the obtained 3D representation was feed to the skeleton tracking system that was able to recognize each skeleton joint of the given human body during the performance of a series of rehabilitation exercises.

Figure 3.1: Overview of the general pipeline. The first stage (I) comprised the acquisition of a 3D representation of the scene using a stereo camera. On the second stage (II), the acquired 3D representation was used to obtain the skeleton configuration of the subject presented within the observed scene.

3.1 Image Acquisition

For the course of this work images were acquired using the stereo camera Bumblebee®2 (BB2-03S2) from Point Grey and FlyCapture® SDK, provided by the manufacturer. Some of the camera's main characteristics are summarized in Table 3.1. The camera possesses a calibration retention system to allow the maintenance of its factory calibration. Both the left and right images are transmitted to a PC via an IEEE-1394 interface with a resolution of 640 x 480 pixels at a maximum frame rate of 48 fps. The provided SDK was used to control the camera settings and acquire the images. A second SDK (Triclops™ Stereo SDK) is also provided by the manufacturer and

offers methods to allow the development of stereo based applications. However, in order to allow the achievement of the maximum frame rate the images were acquired in interleaved format and de-interleaved and rectified in a pos processing stage (Figure 3.2).

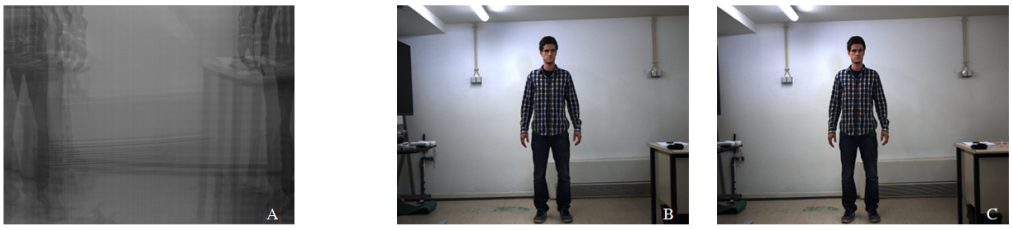

Figure 3.2: (A) Acquired images in interleaved and in (B-C) de-interleaved format, left an right view, respectively. In interleaved format the first byte is from the left camera and the second byte is from the right.

Table 3.1: Specifications of the Bumblebee®2 (BB2-03S2) stereo camera.

Resolution (pixels)	640x480
Pixel Size (µm)	7.4
Maximum Framerate (fps)	48
Baseline (m)	0.12
HFOV (º)	44
VFOV (º)	33
Dimensions (mm)	157 x 36 x 47.4
Mass (grams)	342
Temperature (ºC)	0 to 45

The acquisition protocol is described in detail in Appendix A. Briefly, the stereo camera was placed in a tripod at 0.90 m from the floor and at 2.7 to 3.5 m from the subject. The images were then acquired at a frame rate of about 20 fps with the subject performing a series of rehabilitation exercises.

As the aim is the development of a skeleton tracking system for a telerehabilitation context, during image acquisition the users were instructed to perform the rehabilitation exercises described in Table 3.2 and presented in Figure 3.3. From a biomedical perspective, these three exercises were chosen since they are commonly used in a rehabilitation setting [8, 58]. From a computational perspective, they were chosen since they represent an increasing difficulty for a skeleton recognition system due to its growing complexity. For this reason, the first exercise is consider to be the most simple, the movement occurs in individual planes with none or little occlusions. The third exercise is considered to be the most complex due to the existence of a moment of occlusion of the lower members.

Table 3.2: Detailed description of the rehabilitation exercises performed by the subjects during image acquisition.

Exercise	Description
1	Arm abduction and adduction in the coronal plane followed by arm abduction and adduction in the sagittal plane.
2	Hip abduction and adduction in the coronal plane with the knee extended (left leg followed by the right leg).
3	Toe touch: Movement of the hands from the sides of the trunk in the direction of the toes.

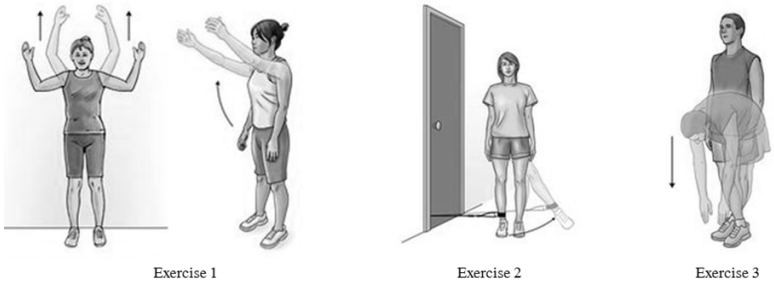

Figure 3.3: Rehabilitation exercises performed by the subjects during image acquisition. (Adapted from [101].)

3.2 Human Body Reconstruction

Before the human pose estimation stage, a 3D representation of the human body to be tracked needs to be obtained. This was achieved by combining color and depth information provided by the stereo camera with the knowledge of the relative position between the views of the stereo pair.

3.2.1 Point Cloud Generation

In stereo based reconstruction the 3D position of a point is found by the intersection of two projection rays from the referred 3D point from two (or more) different views (Figure 3.4) [34]. In order to obtain a 3D model of the scene, stereo systems must deal with two problems, the correspondence and the reconstruction problem [102]. Briefly, the correspondence problem aims to determine the correspondent points in the different views, while the reconstruction problem uses those correspondent points combined with the relative position between the views to obtain the 3D mapping of the scene.

To determine the matching points in the two views the correspondence problem needs to be solved [34], this means detecting for each point in the left image the corresponding point in the right [15]. To simplify the correspondence search, by reducing it to a 1-D search problem on a scanline, rectification needs to be performed. Rectification determines a transformation allowing that pairs of conjugate epipolar lines become collinear and parallel to one of the image axis, usually the horizontal one [102]. After rectification corresponding pairs of points are located within the

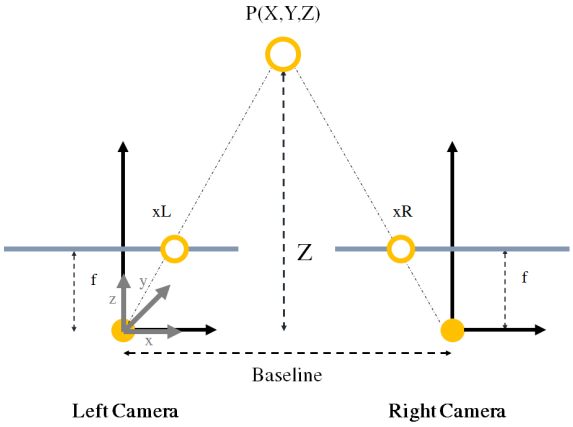

Figure 3.4: Stereo camera geometrical model. The 3D position ($P(X,Y,Z)$) of a point is found by the intersection of two projection rays from the referred 3D point from two different views (xL and xR, respectively for the left and right views). f represents the focal length of each camera and Z the distance between the camera and the 3D point. (Adapted from [34].)

same epipolar line and the difference between their horizontal coordinates results in the so called disparity. As can be observed in Figure 3.5 the rectification corrects camera's misalignments. As well, in the rectification step, camera's distortion was also corrected using the provided camera intrinsic parameters.

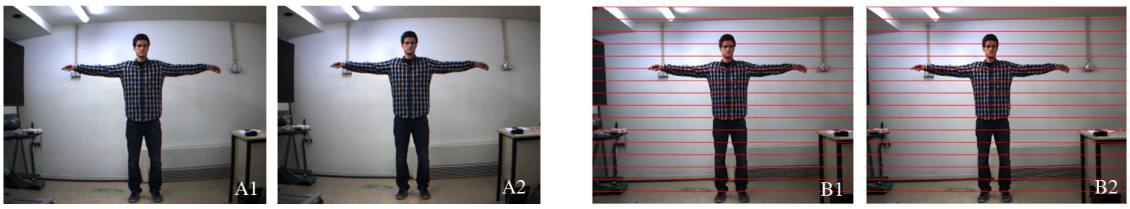

Figure 3.5: (A) Raw de-interleaved images obtained after acquisition. (B) Images after rectification and distortion correction. The red lines represent the epipolar lines. As can be observed, after the distortion correction, the lines (namely the one that marks the junction between the wall and the ceiling) in the image corners are straight instead of curved. (A1), (B1) are images of the left view. (A2), (B2) are images of the right view.

The great majority of the stereo correspondence algorithms follow the structure presented in Figure 3.6. The correspondence detection is accomplished by measuring the similarity between the two points. Many of the used algorithms establish the correspondence between two pixels by using a matching cost function which is aggregated over a window. Following aggregation, the disparity selection is performed and the disparity value is chosen for each pixel. An additional refinement stage is normally performed in order to remove erroneous matches [15]. This refinement stage aims to avoid the existence of peaks, checks for consistency, interpolates gaps and increases the accuracy by performing subpixel interpolation [103].

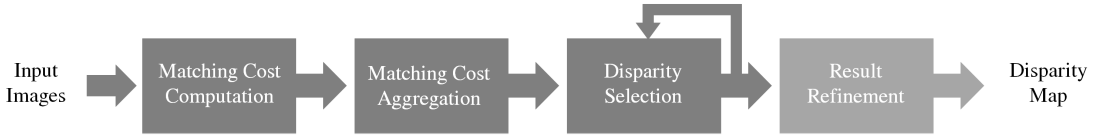

Figure 3.6: Generalized block diagram of a stereo correspondence algorithm. Adapted from [15].

Correspondence search is in fact the most computational burdensome task in 3D stereo reconstruction [34]. Stereo matching algorithms can be grouped into sparse and dense algorithms. Sparse algorithms are obtained by matching features, such as edges or segments, between the stereo images. Dense algorithms can be based on global (energy-based) or local (area-based) methods. The two approaches are counterbalanced by accuracy and speed. Global methods are more accurate and take into consideration the whole image and aim to minimize a global cost function. Local methods, in contrast are faster and use the information given by the intensity points within a predefined window [15]. By producing a dense disparity map, dense algorithms are more suitable for reconstructing surfaces. However, they rely on textured images in order to perform adequately and also are sensible to illumination changes and so are inappropriate when matching image pairs are obtained from very different viewpoints. On the other hand, sparse algorithms despite providing sparse density maps are more robust to illumination changes. Also, by reducing the matches from the entire image to a set of features, sparse algorithms prove to be faster [102].

Three dense stereo matching algorithms were evaluated to solve the correspondence problem:

- Block Matching (BM)
- Semi Global Block Matching (SGBM)
- Variational Block Matching (VAR)

The choice of dense stereo matching algorithms, instead of sparse algorithms, relied on their ability to produce more consistent disparity maps that are more suitable for reconstructing the human body.

The **BM algorithm** [104] is considered to be a local method in which the correspondence between two pixels is established by using a matching cost function which is aggregated over a window. The used matching cost function is the sum of absolute intensity differences (SAD) and the aggregation of costs is accomplished by the superimposition of the cost function (C) over a predefined neighbourhood (S):

$$C(x,y,d) = \sum_{x,y \in S} |\phi_L(x,y) - \phi_R(x+d,y)| \tag{3.1}$$

where $\phi_L(x,y)$ is the gray value of a point (x,y) of the left image, $\phi_R(x+d,y)$ is the gray value of a point of the right image with a disparity of d to the point (x,y).

Finally, the disparity computation (Q) is achieved by selecting the minimum superposition value of the matching costs:

$$Q(x,y,d) = min(C(x,y,d)) \tag{3.2}$$

The **SGBM algorithm** incorporates the advantages of both the global and local classes, achieving a good tradeoff between complexity and quality [17]. The algorithm changes the matching cost function and adds a smoothing item to the energy function to further optimize the final result. A detailed description of the SGBM algorithm can be found in [103]. Briefly, instead of using the Mutual Information (MI) matching cost function proposed by Hirschmuller et al. [103], the used implementation takes advantage of a simpler Birchfield-Tomasi [105] sub-pixel metric. This metric is less sensitive to sampling points and noise whereas the MI function is more robust in relation to recording and illumination changes [106]. The cost aggregation was done by performing an approximation of a global energy function by pathwhise optimizations from 5 directions, instead of the 8 proposed by Hirschmuller et al. [103]. As stated above, an additional smoothness constraint was added to the global energy function:

$$E(D) = \sum_i F(i,D_i) + \sum_{k \varepsilon N_i} P1 * T[|D_i - D_k| = 1] + \sum_{k \varepsilon N_i} P2 * T[|D_i - D_k| > 1] \tag{3.3}$$

where $E(D)$ is the matching cost function sum of all the pixel points; $F(i,D_i)$ is the matching cost of the pixel point i with the disparity D_i. The second and third terms are the smoothing constraints. The second term adds a constant penalty (P1) to the pixels were the disparity changes are small (under 1 pixel). The third term adds a larger constant penalty (P2) for larger disparity changes (above 1 pixel). The use of a small penalty for smaller changes allows an adaptation to slanted or curved surfaces. On the other hand, using a constant penalty for larger changes preserves discontinuities [103]. On the used implementation, the smoothing constraints were set to:

$$P1 = 8 * numberOfImageChannels * SADWindowSize^2, P2 = 4 * P1 \tag{3.4}$$

The disparity computation is then accomplished by a *winner takes it all* approach that is further supported by disparity refinements.

For both the BM and SGBM correspondence algorithms both a pre-processing and a pos-processing stage were conducted. In the pre-processing stage the input images were normalized to reduce lighting differences and to enhance image texture [96]. This was accomplished by running a window (of size 7 x 7) over the entire image. Given the window, the center pixel, I_c, was replaced by $min[max(I_c - \bar{I}, -I_{cap}), I_{cap}]$ where $\bar{I}$ is the average value within the window and I_{cap} is a limit value. During the pos processing stage, three refinements steps were followed after the disparity computation: speckle filtering, consistency check and quadratic interpolation [103, 104].

Speckle filtering aims to remove outliers that result from the absence of texture, reflections and noise. Speckles appear as small patches of disparity that differ consistently from their surrounding disparities. An area threshold value can be set in order to remove these peaks. The consistency check compares the left to right disparity calculation with the right to left calculation within a fixed window. It is particularly useful at range discontinuities, where directional matching will yield different results. Finally, by trying to locate the correlation peak between pixels, by fitting a parabola to the winning cost value and its neighbours [17], the disparity image can be processed to give sub-pixel accuracy. This increases the available range resolution without much additional work [104].

The used **VAR algorithm** is inspired by the one proposed by Kosov et al. [107]. The authors solve the correspondence problem by using a variational method that achieves real-time performance by combining a multi-level adaptive technique with a multigrid approach. The computation effort is reduced by refining the grid only in regions where interesting structures are located. Nevertheless, the used version differs from the one suggested by the authors since the automatic initialization of method's parameters is added, the method of Smart Iteration Distribution (SID) is implemented, the support of Multi-Level Adaptation Technique (MLAT) and the method of dynamic adaptation of method's parameters are not included [108]. In order to reduce the noise a median filter was applied in a pos-processing stage.

The effect of tuning some of the stereo parameters of the three proposed matching algorithms was evaluated and it is presented in Figures 3.7 and 3.8. As already stated the disparity is the difference in pixels between the horizontal coordinates of correspondent points between the two views. The computation time can be reduced by cutting down the number of disparities to be searched which limits the length of a search for a matching point along an epipolar line [96]. However, if the number of disparities is too low the closer objects will not be detected, as can be observed in Figure 3.7 when the number of disparities is set to 16. For this reason, the chosen number of disparities was set to 32 as to be a reasonable compromise between accuracy and processing speed. As for the size of the window used for the SAD calculation, increasing the window size reduces the detail and precision but it is more robust to noise. The opposite effect is observed when smaller windows are used (Figure 3.8). For the stated observations, a size window of 7 was selected as a good counterbalance between the level of noise and detail. As can be observed, the BM algorithm is only able to find strongly matching (high-texture) points between the two images and so it is not the most adequate for an indoor scene were some low textured large areas, such as walls, can be found. The VAR algorithm was not able to return suitable results as can be observed by the black dots that appear on the disparity maps. As well, the subject borders are not as defined as in the previous two methods. From the three evaluated stereo matching algorithms, the SGBM was the one that returned the most reliable disparity map and so it was the one chosen to carry on the processing pipeline.

An extensive overview of stereo matching algorithms developed in the last years can be consulted in [15] and a ranking chart according to the taxonomy of Scharstein and Szeliski [109] is

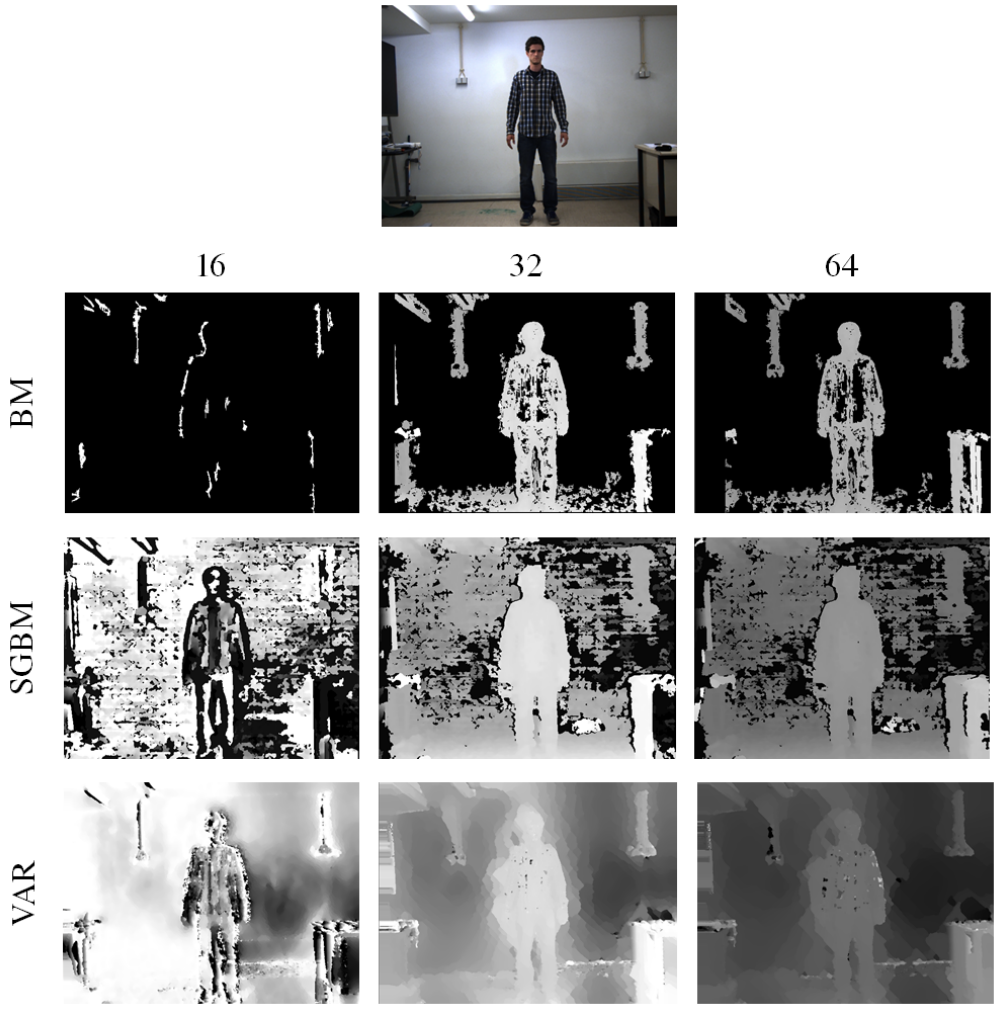

Figure 3.7: Disparity maps obtained using the three proposed stereo matching algorithms. Number of disparities was varied from 16 to 64. All the other parameters were set constant. Black pixels represent unknown disparity values. Brighter pixels represent points with largest disparities and so closer to the camera. The correspondent RGB reference image (left view) is presented on the top row for comparison.

availabe online[1].

After finding correspondent points in the two views the 3D scene can be obtained by using triangulation. This process can only be successful if the relative position of the two cameras, such as their rotations and translations, as well as intrinsic camera characteristics, such as the focal length or the principal points are known. This parameters, called the extrinsic and intrinsic parameters, respectively, are determined during camera calibration [34]. Camera calibration can be performed using epipolar geometry. The principles behind epipolar geometry are extensively explained in [110]. A wide variety of calibration tools, such as the Camera Calibration Toolbox for Matlab[2], can be used to obtain the mentioned parameters. However, the used stereo camera

[1] `http://vision.middlebury.edu/stereo/eval/`
[2] `http://www.vision.caltech.edu/bouguetj/calib_doc/`

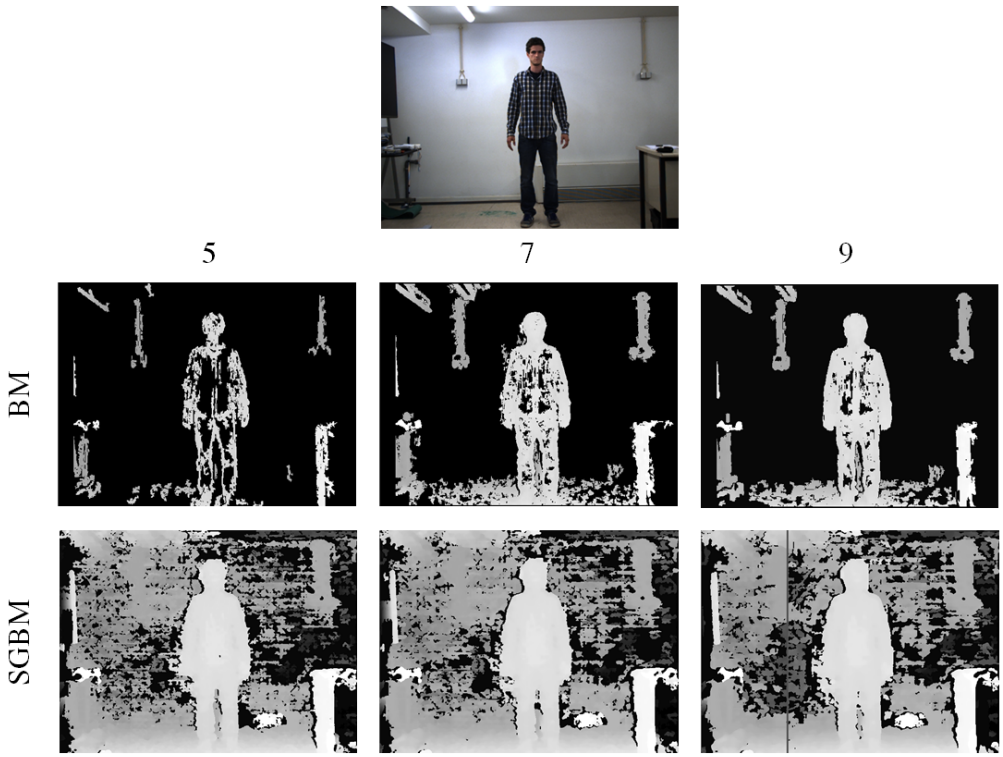

Figure 3.8: Disparity maps obtained using the BM and SGBM stereo matching algorithms. The size of the window used for the SAD calculation was varied from 5 to 11 pixels. All the other parameters were set constant. Black pixels represent unknown disparity values. Brighter pixels represent points with largest disparities and so closer to the camera. The correspondent RGB reference image (left view) is presented on the top row for comparison.

already provides factory calibrated stereo parameters that are stored in the device's memory. These parameters are summarized in Table 3.3 and were used throughout this work.

Table 3.3: Camera calibration parameters of the Bumblebee2 stereo camera.

f_x	f_y	c_x	c_y
800.3968	800.3952	323.155	242.366

According to the pinhole camera model, the camera parameters can be summarized in the projection matrix that is used to estimate the world coordinates $P(X,Y,Z)$ from the pixel coordinates $p(u,v)$ [56]:

$$w\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & s & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & t_1 \\ r_{21} & r_{22} & r_{23} & t_2 \\ r_{31} & r_{32} & r_{33} & t_3 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{3.5}$$

where (f_x, f_y) are the pixel-related focal lengths representing the actual focal length f on the image coordinate system, (c_x, c_y) are the central points in pixel coordinates, s is the skew factor, w is the scaling factor, R is the rotation matrix and t is the translation vector. For the used stereo system, the pixels are considered to be squared and so the skew is zero. Also, the two views are parallel in the X axis, with a translation of 0.12 m (that corresponds to the stereo camera baseline), and with no rotation between then and so equation 3.5 becomes:

$$w\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0.12 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \tag{3.6}$$

The 3D point cloud of the scene was obtained by combining the disparity values previously obtained with the parameters of Table 3.3 and the following equation:

$$\begin{bmatrix} X/w \\ Y/w \\ Z/w \\ 1 \end{bmatrix} = Q \begin{bmatrix} u \\ v \\ d(u,v) \\ 1 \end{bmatrix} \Leftrightarrow \begin{bmatrix} X/w \\ Y/w \\ Z/w \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & c_x \\ 0 & 1 & 9 & -c_y \\ 0 & 0 & 0 & f \\ 0 & 0 & -1/t_1 & (c_x - c_x^{'})/t_1 \end{bmatrix} \begin{bmatrix} u \\ v \\ d(u,v) \\ 1 \end{bmatrix} \tag{3.7}$$

where $d(u,v)$ is the disparity value at the location (u,v) and Q is the perspective transformation matrix that represents the disparity-to-depth mapping.

Nevertheless, as the point clouds were very noisy a segmentation and denoising step was followed and is described in detail in the following section.

3.2.2 Point Cloud Segmentation and Denoising

In order to improve the raw point clouds obtained directly after disparity computation and triangulation, a segmentation followed by a filtering stage was implemented.

As can be seen in Figure 3.9, the raw point cloud possessed different kinds of noise. Due to the nature of the chosen algorithm in the stereo correspondence stage the foreground pixels are propagated in the background direction and so there is not a clear definition between the foreground and the background. This lateral noise, that is located around the subject, results in noisy borders with points that notably belong to the background. To surpass this situation a segmentation methodology was applied.

Also, since the background is a white textureless wall the stereo correspondence algorithm performs poorly and so it is not able to estimate the depth accurately. To improve the quality of the background a plane fitting method was followed.

The foreground segmentation was accomplished by combining 2D with 3D information [111]. The used pipeline is described in Figure 3.10. Briefly, the disparity image is used to obtain a rough

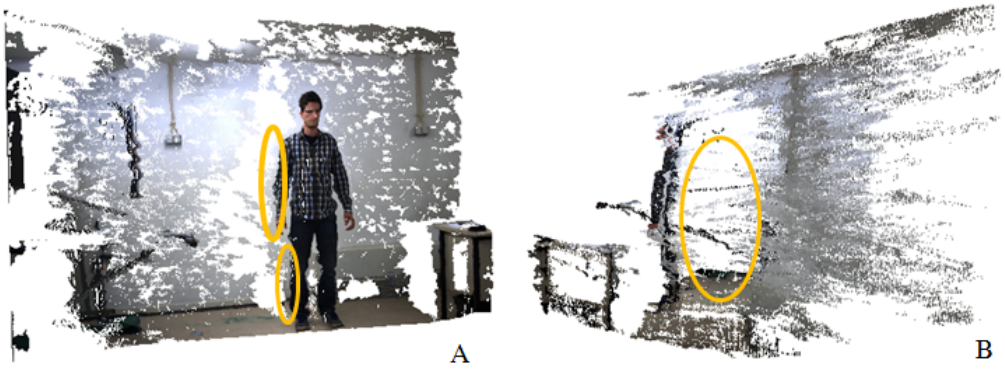

Figure 3.9: Two different perspectives of a raw point cloud obtained directly after disparity computation and triangulation. Yellow circles highlight the presence of lateral noise.

estimate of the subject's position and, after some refinements, combined with the RBG information to obtain the final segmentation result by using the GrabCut method [112].

Given the relatively simple image setup with only the subject in the line of sight of the camera and a wall behind him, the subjects position was estimated by applying the Otsu's [113] binarization method on the disparity image (Figure 3.10A-B). Since in a disparity image each pixel is inversely proportional to the distance from the camera, the same object has similar disparities and the background has lower disparity values normally. For this reason, the histogram of the disparity image will present two distinct peaks, one for the subject and the other for the background (in this case the wall). By using the OTSU's method the disparity value that allows the separation between this two peaks (that is located in the valley) can be automatically extracted. This initial mask could have been obtained by using other methods, such as background subtraction. However, disparities are more robust in relation to illumination and shadows that can have hampering effects in background subtraction like algorithms. Nevertheless, the presence of low texture, repetitive patterns, reflections, noise and occlusion can result in completely wrong disparities. In order to reduce the effect of this kind of noise an erosion morphological operation was applied to the binary image (Figure 3.10B-C). The pixels of the binary image were then marked as probable foreground and the black pixels as background and combined with the RGB image to proceed with the segmentation by using the GrabCut method. The GrabCut method is extensively described in [112]. Briefly, this method is based on color Gaussian Misture Model and an iterative energy minimization optimized by using the graph cut algorithm. As a result, the binary mask of Figure 3.10E is obtained. However, due to the existence of white squares in the shirt of the subject the returned mask presented some holes. For this reason an hole filling methodology was applied. Also, assuming that the object of interest (in this case the subject) was the bigger blob, all the other blobs were removed, obtaining the final mask presented in Figure 3.10F. The final mask was used to obtain the refined RGB segmented subject presented in Figure 3.10H that was projected to 3D (Figure 3.10I).

In a textureless background, as can be observed in Figure 3.9, the obtained raw point cloud

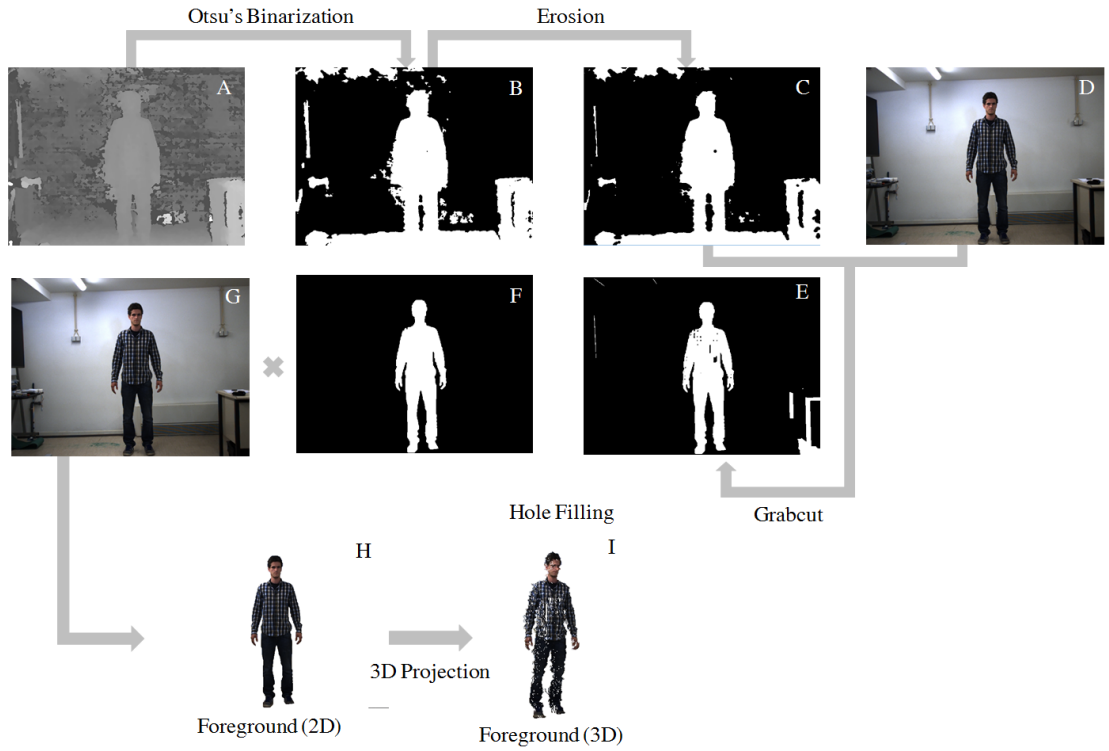

Figure 3.10: Foreground segmentation pipeline. The disparity image (A) is used to obtain a rough estimate of the subject's position (B) through Otsu's binarization, that is further improved by using an erosion morphological operation (C). The obtained mask is then combined with the RBG information (D) to obtain the segmentation result by using the GrabCut method (E). The retrieved mask is then corrected (F) and used to obtain the refined RGB segmented subject (H) that was projected to 3D (I).

presents an high amount of wrong calculated depths. For this reason, a plane fitting methodology was followed in order to obtain an estimation of the wall and floor planes, Figure 3.11.

In order to roughly separate the points that belonged to the floor from the ones that belonged to the wall a passthrough filter was applied on the input cloud (Figure 3.11C-D). This was done by removing all the points with Z dimension outside the range of 0 to 4.7 m, for the case of the floor, and outside the range of 4.0 to 5.0 m for the case of the wall. This range values were empirically selected. In the future an automatic range selection should be implemented, based, for example, on the analysis of the histogram of the disparity map.

The plane fitting was then followed using the RANdom SAmple Consensus (RANSAC) method [114] supported by the calculation of surface normals, in each one of the two point clouds previously obtained.

The surface normal $\vec{n}$ of a point p was estimated by the normal to the k-neighbourhood surface by performing PCA on the neighbourhood's covariance matrix. The direction of $\vec{n}$ was obtained by the eigenvector corresponding to the smallest eigenvalue. For each point p_i a covariance matrix C

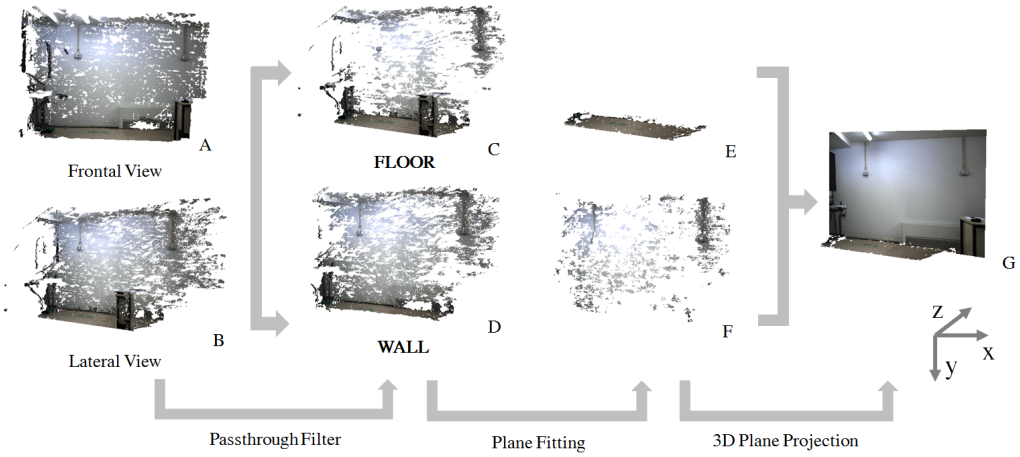

Figure 3.11: Plane fitting pipeline. The raw background cloud was roughly segmented into the wall and floor clouds by applying a passthrough filter in the Z dimension (A-D). A plane model fitting methodology was then applied to each one of the previously separated clouds in order to estimate the floor and the wall planes (E-F). The plane fitting was accomplished by using the RANSAC method supported by the calculation of surface normals. After obtaining the plane coefficients the points from the initial cloud were projected to 3D, resulting in the final refined background cloud (G).

is assembled as follows:

$$C = \frac{1}{k}\sum_{i=1}^{k}.(p_i - \overline{p}).(p_i - \overline{p})^T, C.\overrightarrow{v_j} = \lambda_j.\overrightarrow{v_j}, j \in \{0,1,2\} \tag{3.8}$$

where k is the number of point neighbours (here equal to 50) considered in the neighbourhood of p_i, $\overline{p}$ represents the 3D centroid of the nearest neighbours, λ_j is the j-th eigenvalue of the covariance matrix and $\overrightarrow{v_j}$ the j-th eigenvector [115].

After the normals calculation the plane model was fitted using RANSAC. This algorithm is extensively described in [114]. Briefly, considering that the goal is to estimate a plane:

1. the algorithm begins by randomly selecting three points from the input cloud and calculating the correspondent plane parameters;
2. according to a given threshold (here set to 0.15 for the wall plane and 0.10 for the floor plane), all the points from the original cloud that belonged to the calculated plane are selected;
3. steps 1 and 2 are repeated N times (here N was set to 100). In each iteration the previous result is compared to the new one, by estimating the error of the inliers in relation to the model, and replaced if better. Or until a confidence of 99% is found.

Since the RANSAC algorithm is supported by the computation of normals a second threshold was considered in step 2. This threshold sets the relative weight (between 0 and 1) to give to the angular distance (0 to $\pi/2$) between point normals and the plane normal. Here, this threshold was considered to be 0.1.

As can be observed in Figure 3.11E-F the output of the RANSAC plane model fitting is a set of inlier plane points and the plane parameters that represent each of the fitted planes. The obtained

plane parameters and inlier plane points were used to produce the final background cloud. This was accomplished by projecting the inlier points that belonged to the wall cloud to the wall plane and the ones that belonged to the floor cloud to the floor plane. The points that were in the initial raw cloud, but were not in the inlier plane points returned by the RANSAC algorithm were projected to the wall plane. For each point (u,v) in the 2D image, the projection to the 3D point (x,y,z) was done based on the following relation:

$$ax+by+cz+d=0 \Leftrightarrow z=\frac{-d}{\frac{a(u-c_x)}{f_x}+\frac{b(v-c_y)}{f_y}+c} \tag{3.9}$$

where a,b,c and d are the plane coefficients, f_x and f_y are the pixel-related focal lengths representing the actual focal length f on the image coordinate system and (c_x,c_y) are the central points in pixel coordinates. The x and y coordinates were then calculated according to the following equations:

$$X=\frac{(u-c_x)*Z}{f_x} \tag{3.10}$$

$$Y=\frac{(v-c_y)*Z}{f_y} \tag{3.11}$$

Before combining the background and the foreground into a single point cloud, the foreground information was smoothed using a bilateral filter. This filter was chosen since it is non-iterative (meaning that it achieves good results with only one single pass), preserves edges and is fairly simple [116]. The bilateral filter combines range (intensity) with domain (spatial) information. The filter replaces each pixel by a weighted average of its neighbours (Figure 3.12). The weight of a pixel depends on a function G_{σ_s} in the space domain (S), that decreases the weight of pixels with large distances in the image plane, and on a function G_{σ_r} in the intensity domain (R), that decreases the weight of pixels with large intensity differences. Using a Gaussian G_σ as the decreasing function and considering an image I, the result I^{bf} of the bilateral filter for the pixel p is defined by:

$$I_p^{bf}=\frac{1}{W_p^{bf}}\sum_{q\varepsilon S}G_{\sigma_s}(||p-q||)G_{\sigma_r}(|I_p-I_q|)I_q \tag{3.12}$$

with,

$$W_p^{bf}=\sum_{q\varepsilon S}G_{\sigma_s}(||p-q||)G_{\sigma_r}(|I_p-I_q|) \tag{3.13}$$

where the parameter σ_s defines the size of the kernel in the spatial neighbourhood used to filter a pixel, and σ_r controls how much an adjacent pixel (q) is downweighted because of the intensity difference ($|I_p - I_q|$). W_p^{bf} normalizes the sum of the weights [116].

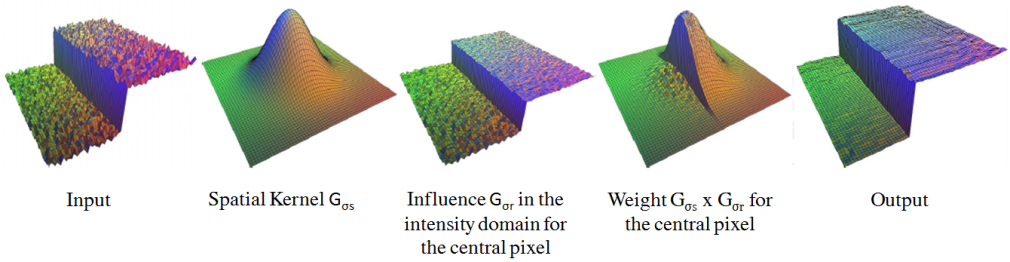

Figure 3.12: Bilateral Filtering. Colors are used to give the notion of shape. (From [117].)

By acting as a standard domain filter, the bilateral filter removes small and weakly correlated differences between pixel values caused by noise. As well, due to the range component of the filter, edges are preserved.

The filter was tested using different parameters, varying the spatial kernel (σ_s) and the range (σ_r) kernel, independently. When the spatial kernel was varied, the range kernel was kept constant and equal to 0.1. When the range kernel was varied, the spatial kernel was kept constant and equal to 5.0. The results are presented in Figure 3.13 and Figure 3.14, respectively. The results were only assessed trough visual comparison. When the spatial kernel is varied (from 1.0 to 15.0), an accentuated smoothing is observed between $\sigma_s = 1.0$ and $\sigma_s = 5.0$. Nevertheless, when the size of the spatial kernel is further increased there are no notable differences. So, a size of 5.0 for the spatial kernel was considered to be adequate. Regarding the size of the range kernel (σ_r), when it is above 1.0, the smoothness is excessive with a considerable loss of detail. For this reason, the value of 0.1 was considered to be the best compromise between an adequate smoothing without loss of detail.

Finally, the smoothed foreground was combined with the refined background resulting in the final cloud that is given to the skeleton tracking system described in detail in the next section (Figure 3.15). For the developed system the background cloud is acquired once in the beginning of the acquisition and only the foreground is updated.

3.3 Human Pose Estimation and Motion Tracking

The problem of "skeleton tracking" or "markerless motion capture" can be found in a wide range of applications, from the detection of bounding boxes around a person to fully articulated body models [118]. The level of detail and precision is directly related to the envisioned application. In a telerehabilation setting the aim is to extract clinically relevant information from the patient while he performs a set of prescribed exercises. The skeleton tracking system needs to be the most

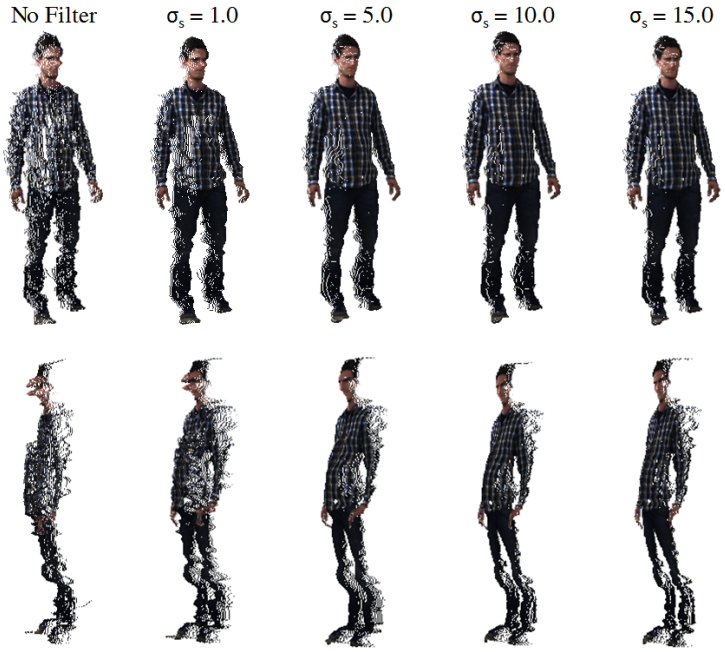

Figure 3.13: Effect of the variation of the σ_s parameter of the bilateral filter. This parameter represents the kernel in the spatial neighbourhood used to filter a pixel. Here, the σ_r parameter was kept constant and equal to 0.1. Point clouds are presented in frontal (top row) and lateral (bottom row) view.

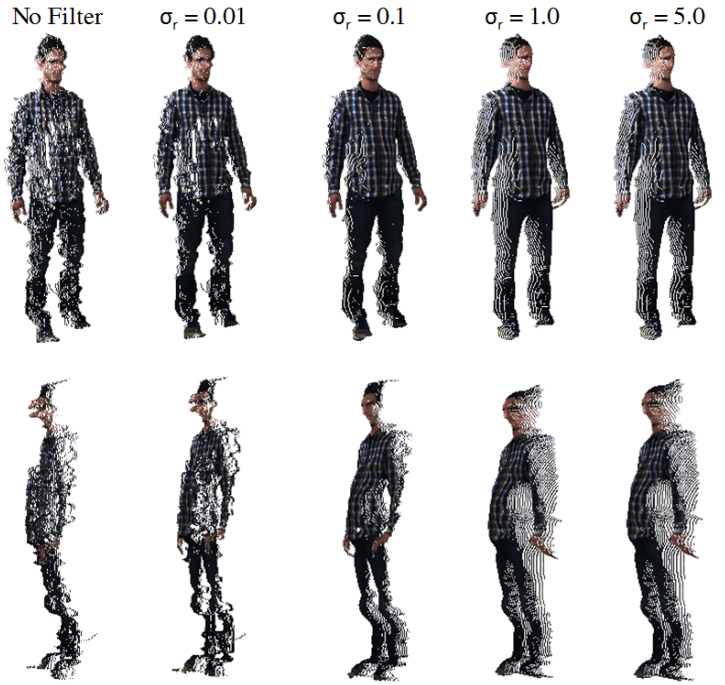

Figure 3.14: Effect of the variation of the σ_r parameter of the bilateral filter. This parameter controls how much an adjacent pixel is downweighted because of the intensity difference. Here, the σ_s parameter was kept constant and equal to 5.0. Point clouds are presented in frontal (top row) and lateral (bottom row) view.

Figure 3.15: After the described filtering process, the plane fitted background is combined with the foreground.

detailed and precise possible in order to support the robustness of the collected data. As well, the real-time applicability should not be forgotten.

One of the most widely spread approaches for skeleton tracking is the one proposed by Shotton et al. [84] that is provided with the Microsoft® Kinect SDK. Nevertheless, this approach is *closed* and can only be used if a Microsoft® Kinect device is connected to the computer. Also, due to the commercial nature of this hardware, the complete description of the underlying recognition approaches cannot be found in the scientific literature [9]. Another fairly spread method can be found on OpenNI [119], but like the one provided by the Microsoft® Kinect SDK, it can only be used with PrimeSense devices.

Since the goal of the present work was to explore the use of a non active method to acquire the 3D information, namely a stereo camera, the previous methods could not be used. Nevertheless in September 2014, PCL released a new module called GPU People [120] that included some methods to perform skeleton tracking that work with PCD data, being device independent. Despite not being yet in a fully mature state, that implementation served as a stepping stone for the present work. Also, since the mentioned library has an open architecture and the obtained results were not the most stable the provided implementation was improved. The used implementation is based on the work developed by Koen et al. [118] that is inspired by the one of Shotton et al. [84] in the sense that both use pixel-wise body part labelling in order to retrieve body part proposals. However, the first does not requires background subtraction an can therefore be used with a non-static camera. The already implemented and the improved methodology is described in detail in the next sections.

3.3.1 Pixel-wise Body Part Labelling

The pixel-wise body part labelling was accomplished by training a RDF [85] classifier that is able to attribute body part labels to each image pixel.

The RDF was trained by generating synthetic data. This synthetic data was obtained by mapping real motion capture data onto a virtual model of a person (Figure 3.16). The virtual model was the one of a single slim male of Make-Human (MH)[3] that consists of a parametrically mor-

[3] `http://www.makehuman.org/`

phed (by age, height, etc.) kinematic chain covered by a mesh structure. The process of creating the virtual model is further described in [121, 122]. The real motion capture was obtained from the CMU MoCap database[4] . This dataset contains a large set of varied poses captured by a Vicon system at 120 Hz. In order to reduce the size of the dataset, poses that were not directly related to daily activities, such as break dancing, were removed. As well, due to high frame rate acquisition many poses were repeated and so a greedy sparsification was applied ensuring at least 5 degrees of movement in joint angles. This process resulted in a final number of 80k poses. Next, to map the kinematic chain of the used dataset onto the MH labelled model and to render the annotated depth images Blender[5] and OpenGL[6] were used. It is noteworthy that the annotated depth images were generated considering a straight-on, chest height camera angle. The used methodology is described in detail in [16].

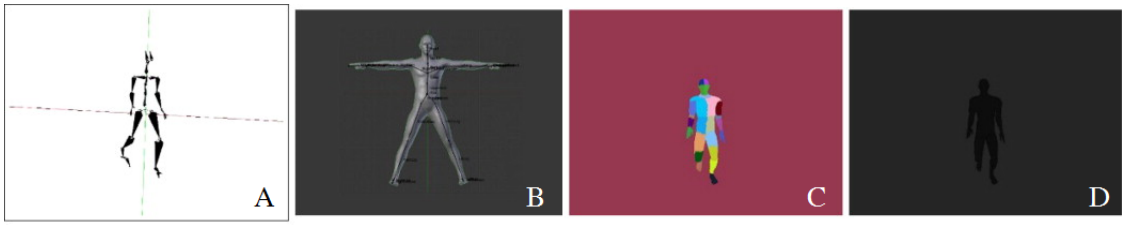

Figure 3.16: Generation of synthetic data to train the RDF. (A) The MoCap data is mapped onto a (B) 3D graphics body model. The body part labels are annotated resulting in a (C) body-part labelled model and the corresponding (D) depth image. (Adapted from [118].)

After obtaining the training data, the RDF was learned by acquiring depth image features (as in [84, 118]). For a pixel x in the image I, a feature is the difference in depth between two randomly chosen points:

$$f_\theta(I,x) = d_I(x + \frac{o_1}{d_I(x)}) - d_I(x + \frac{o_2}{d_I(x)}) \tag{3.14}$$

where $d_I(x)$ is the depth of x, and $\theta = (o_1, o_2)$ is a pair of pixel offsets which are normalized by depth (in order to insure depth invariance). During training, the described features were computed for pixels on the body and discretized into 50 uniform bins. A large depth value was given to pixels that lied on the background. Figure 3.17 illustrates the pixel classification approach. Feature f_{θ_1} points upwards: for pixels x near the top of the body, Equation 3.14 will return a large positive response, but for pixels x near the lower part of the body the returned value will be close to zero. On the other hand, feature f_{θ_2} may help distinguish vertical structures such as the arm [84].

The feature vector $f_\theta(I,x)$, for each pixel, contains 2000 features obtained by randomly sampling offset pairs. Meaning that each feature vector was calculated for 2000 randomly chosen pixel locations. The obtained feature vectors were used to learn the RDF. A forest is a set of T decision

[4] http://mocap.cs.cmu.edu/
[5] https://www.blender.org/
[6] https://www.opengl.org/

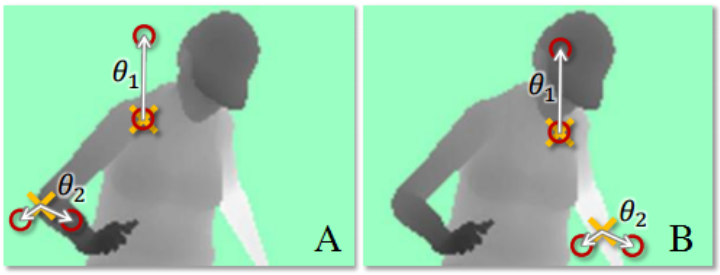

Figure 3.17: Depth image features used for pixel-wise body part labelling. Pixels being classified are indicated by the yellow crosses. The offset pixels indicated in Equation 3.14 are marked by the red circles. (A) The two example features give a high depth difference response. (B) The same two features considered in new image locations result in a much smaller response. (From [84].)

trees, each consisting of a split and leaf nodes (Figure 3.18). A feature f_θ and a threshold τ are found in each split node. In order to proceed with the pixel labelling, one starts at the root and iteratively evaluates Equation 3.14, choosing the left or right leaf according to the threshold τ. The feature vectors with their ground truth labels, $f_\theta(I,x,c)$, were used to learn the RDF. The forest estimate, $P(c|I,x)$, was obtained by combining each of the posterior distribution over the pixel label, $P_t(c|I,x)$, that are produced by each one of the N_t trees in the forest:

$$P(c|I,x) = \frac{1}{N_t}\sum_{t=1}^{N_t} P_t(c|I,x) \tag{3.15}$$

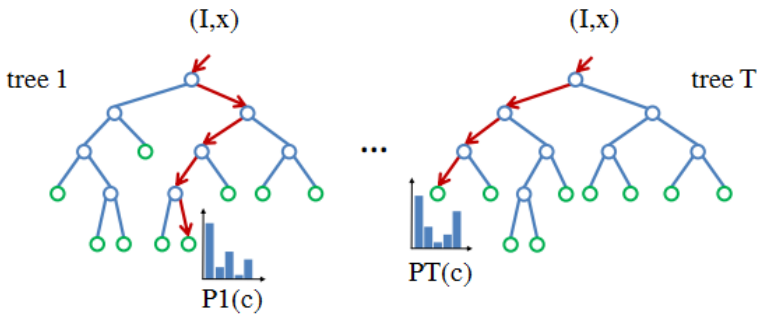

Figure 3.18: Randomised decision forests used for pixel-wise body part labelling. A forest is a set of T decision trees, each consisting of a split (blue) and leaf (green) nodes. Red arrows show different paths that can be taken by different trees for a particular input. (From [84].)

All the pixels, including the background, were labelled. As to reduce the noise from the initial classification, a mode blur filter (with a 5x5 window) thresholded by a depth value of 0.3 m was used [118].

As described in detail in [118], the RDF learning was done by formulating it as a MapReduce [123] problem. The generation of the training data and the learning of the RDF for the 80k poses took seven days to be completed.

The result of the RDF learning was a set of decision trees each with 20 levels and so containing a bit more than a million leaf nodes. According to Equation 3.15 the most likely body part label was assigned to each pixel. In the case that the resultant probability distribution does not produce a maximum likelihood estimation, the label of the current pixel was chosen accordingly to the most consistent label given to the eight neighbouring pixels [16].

The system was prepared to receive up to four decision trees and already provided three trained decision trees. As supported by Figure 3.19, as the number of used trees decreases the labelling outcome deteriorates. For this reason, the output data was generated considering the use of three trees. Nevertheless, the provided trees by PCL are prepared for Kinect like data. This means that they only work when the given PCD data is obtained from a camera with the intrinsic characteristics, such as the focal length, equal to the ones of the Microsoft® Kinect. For this reason, the provided trained data was adapted to work with the data given by the used stereo camera. This was accomplished by up scaling the offsets of the features of the existing trees according to the Bumblebee® 2 focal length.

Figure 3.19: Effect of the number of trees on the labelling outcome. As can be observed (from C to A) a larger number of used trees produces a less noisier labelling outcome.

3.3.2 Skeleton Estimation

After the pixel-wise body part labelling, each pixel has an associated label. To extract a valid skeleton, the ensemble of pixel labels, $C = \{c(x)\}$, must be clustered into a smaller set of body part proposals V. The clustering was accomplished by a breadth-first search over all the connected pixels with the same label within a given distance threshold in 3D. These labels were kept if their size was above a predefined number of pixels (here set to 200). After this clustering process the centroid and covariance were calculated for each cluster. From these, the eigenvectors and eigenvalues were calculated. Based on feasible anthropometric values for each body part length, the first eigenvalue (largest dimension) of each cluster was evaluated and maintained (or not) as a body part candidate [118]. At this point a set of body part proposals, V, has been generated. From these proposals a set of skeleton configurations, S, can be obtained:

$$P(V|C,I) \rightarrow P(S|V) \tag{3.16}$$

The obtained clusters were evaluated based on kinematic constraints in two phases. Initially, the clusters were organized according to their size and label. The local consistency between all links was assessed and considered to be valid if each pair of body parts had a parent-child or grandparent-grandchild relationship in the kinematic model. This evaluation was performed according to the kinematic relationship of the used skeleton model (Figure 3.20). Secondly, the obtained locally consistent pairs were used to obtain globally consistent skeletons. A skeleton tree candidate is rooted in the neck since this part was empirically found to be the most stable. Then, all the globally consistent skeleton proposals are evaluated based on the global error, the normalized error and the number of found body parts in each skeleton candidate [118].

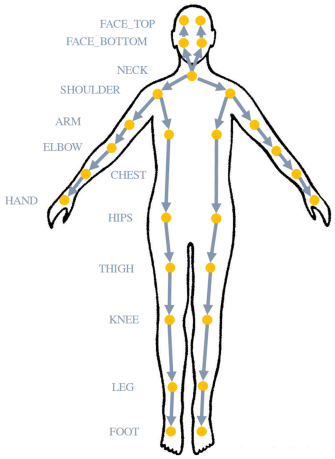

Figure 3.20: Used kinematic skeleton model. The arrows indicate the order in which a parent is connected to its child. (Adapted from [118].)

Nevertheless, the obtained skeleton candidates were, sometimes, incomplete or mislabelled due to noise, occlusions or clutter. In order to improve the skeleton estimates the RGB and depth information were combined in an online appearance model estimation and segmentation (Figure 3.21). Based on the initial segmentation of a person, a model of their location in space and their appearance as given by the hue in the Hue, Saturation, Value (HSV) colour space was learned. The new segmented pixels were then fed to the process of part estimation and skeleton extraction resulting in more robust estimates [118].

Despite the improvement of the labelling process provided by the online appearance model estimation, as mentioned by Alina et al. [124], in some situations the implemented system had some problems with large surfaces specially the floor and the walls. To solve this problem, the Ground Plane Detector [125, 126] was incorporated and used to segment the body cluster prior to the labelling. For this, three points of the ground plane were selected, after which the Ground Plane People Detector removed the ground plane and estimated the point cloud belonging to the human. The points of the cluster were then transformed to the depth image, setting all other depth pixels to very high values [124]. As presented in Figure 3.22, the use of the people detector improved the labelling outcome.

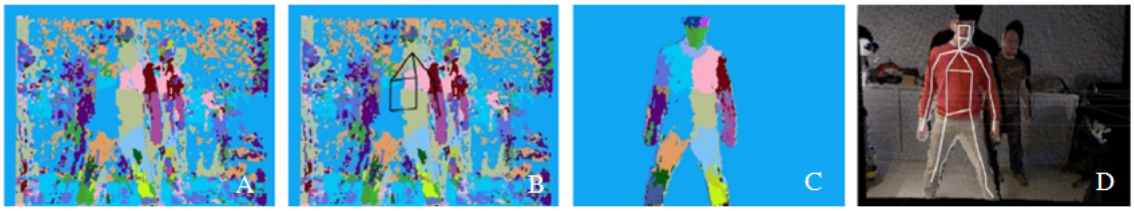

Figure 3.21: Body part labelling refinement based on online appearance model estimation. (A) The original labelling is very noisy an results in an inaccurate skeleton estimate (B). Based on the initial labelling, an online estimation of a colour and depth based appearance model leads to a much cleaner labelling (C) from which a more reliable skeleton candidate can be extracted (D). (From [118].)

Figure 3.22: Labelling outcome for the same frame considering the use of the people detector (B) and without the people detector (C). Without the use of the detector the system is not able to correctly identify the people position. The correspondent (A) RGB reference image is presented on the top row for comparison.

3.3.3 Joints Position Correction

The returned position of some joints was not stable and so their position needed to be reviewed and corrected. These joints were the shoulders, elbows, hands, hips, thighs, knees and legs. The joint revision and correction was performed given the kinematic tree presented in Figure 3.20. For the lower members the followed correction order was hips, thighs, knees and legs and for the upper members shoulders, elbows and hands. This orderly correction was performed since in most cases joint position evaluation and correction is based on the previous joint in the kinematic chain.

The implementation in [124] only considered the correction of the shoulders, elbows and hips. The shoulders correction implementation returned stable results and so it was not improved. The elbows and the hips were not steady and so their calculation was enhanced. In the cases were a previous implementation was improved, the previous used algorithm is presented for comparison.

Some of the centroids returned do not correspond to real joints, such as the thighs or the legs. During this work the term “joints” will also be applied to those cases. A joint position is corrected when its distance to its parent joint is not anthropometrically valid. Valid body parts lengths expressed as a percentage of body height are presented in Figure 3.23B. Table 3.4 contains anthropometrically feasible lengths between joints and its children and its information was derived from the information given by Figure 3.23B, considering that the points of interest are located in the middle point of the provided lengths. For example, if the intention is to evaluate the shoulder to arm length, since the provided distance is between the shoulder and the elbow (0.189H, were

H is the person's height), the shoulder to arm distance can be obtained by considering half of the provided distance in Table 3.4 (0.189H/2 = 0.095H). The lengths provided by Table 3.4 were used as an overall guidance for anthropometrically valid distances between joints. Due to the variability of the human body and also since the used information is dependent on the person height (that was automatically calculated and so it is prone to errors) a confidence level of +- 30% was considered for the assessment of the distance between joints. The confidence level was empirically chosen.

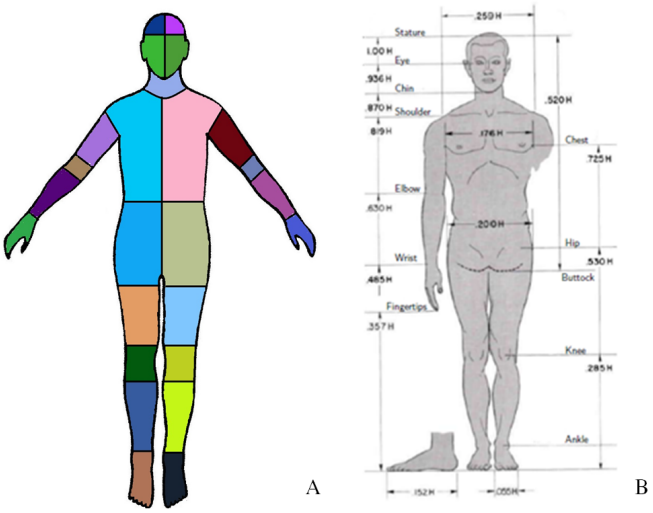

Figure 3.23: (A) Ground-truth body part label model used for the determination of each skeleton joint and body part positions. (B) Body segments length expressed as a percentage of body height (H) for a US Male. (From [127].)

A second option was also tested in order to determine anthropometrically valid distances between joints. Instead of using predefined values found in literature [127], the body part lengths were determined assuming an initial calibration position. From this calibration position body lengths specific to each user were retrieved online and stored for evaluation in subsequent frames. An initial pose calibration should be in frontal position looking within the camera direction with legs apart and arms not touching the torso. This initial pose was chosen since the labelling performs well in this conditions. This is due to the fact that this pose presents no occlusions. Also, when some parts of the body touch each other, such as the hands and the torso, or both legs, the system has a bigger difficulty in labelling each body part.

After the correction implementation each retrieved joint has an associated state. "Inferred" if the joint position was found to be invalid and so it was corrected or "not inferred" if the joint position given by the centroid of the blob was considered valid. The implemented corrections are described bellow.

Table 3.4: Anthropometrically feasible lengths between the body parts/joint locations and its children expressed as a percentage of total height. Example of interpretation: Lshoulder has two children, Larm and Lchest, which should be located at a distance of 0.160H and 0.095H meters, respectively. H – Height.

Father	Child				Father	Child			
	1st	2nd	3rd	4th		1st	2nd	3rd	4th
Lfoot	—	—	—	—	**Rhand**	—	—	—	—
Lleg	0.145	—	—	—	**Larm**	0.100	—	—	—
Lknee	0.123	—	—	—	**Lelbow**	0.073	—	—	—
Lthigh	0.123	—	—	—	**Lforearm**	0.137	—	—	—
Rfoot	—	—	—	—	**Lhand**	—	—	—	—
Rleg	0.145	—	—	—	**faceLB**	0.064	—	—	—
Rknee	0.123	—	—	—	**faceRB**	0.064	—	—	—
Rthigh	0.123	—	—	—	**faceLT**	—	—	—	—
Rhips	0.125	—	—	—	**faceRT**	—	—	—	—
Lhips	0.125	—	—	—	**Rchest**	0.210	—	—	—
Neck	0.080	0.080	0.085	0.085	**Lchest**	0.210	—	—	—
Rarm	0.100	—	—	—	**Lshoulder**	0.160	0.095	—	—
Relbow	0.073	—	—	—	**Rshoulder**	0.160	0.095	—	—
Rforearm	0.137	—	—	—					

Shoulder Calculation

As can be observed in Figure 3.23A, the shoulders region do not possess their own blob and so the shoulders position was inferred based on the position of the chest blob (Algorithm 1). Considering the evaluated movements, the algorithm in [124] returned stable results and so it was not improved.

Algorithm 1 Shoulders Calculation

Get the highest point of the chest blob $(X_{max}, Y_{max}, Z_{max})$.
Apply a passthrough filter on the chest blob to remove all the points with Y dimension outside Y_{max} +- 0.10 m.
Calculate the centroid of the resulting blob, which will yield the shoulder position.

Elbow Calculation and Correction

In [124], the elbow position was only re-calculated when the elbow blob was missing or if it was too small (under 200 pixels) to be considered as a valid blob (Algorithm 2).

Algorithm 2 Elbow Calculation (old version)

Get the point $(X_{max}, Y_{max}, Z_{max})$ with the maximum distance from the shoulder joint.
Apply a passthrough filter, on the arm blob, to remove all the points with Y dimension outside Y_{max} +- 0.05 m.
Calculate the centroid of the resulting blob, which will yield the elbow position.

However, as can be seen in Figure 3.24A1 sometimes the re-calculation fails when the arm blob has a wavy border near the forearm blob. In this situation the estimated elbow position was not the most reliable and so a more robust algorithm was implemented in order to correct a wider range of circumstances.

On the new implementation (Algorithm 3), the calculation was considered not only when the elbow blob was missing or it was too small, but also when the elbow blob was found. The latter calculation was taken into consideration since, due to the small dimensions of the elbow blob, sometimes its position was not accurate. On the top of the calculation, a correction (Algorithm 4) was introduced to insure that the arm to elbow and elbow to forearm distances were anthropometrically valid.

Algorithm 3 Elbow Calculation (new version)

if the elbow blob is missing or it is too small (situation 1) **then**
Get the point $(X_{max}, Y_{max}, Z_{max})$, from the arm blob, with the maximum distance from the shoulder.
Fuse the arm with the forearm blob (arm-forearm blob).
Apply a passthrough filter, on the arm-forearm blob, to remove all the points outside a square of 0.05 m around the $(X_{max}, Y_{max}, Z_{max})$ point.
Calculate the centroid of the resulting blob, which will yield the elbow position.
else if the elbow blob is found (situation 2) **then**
Use the centroid of the elbow blob as a first proposal for the elbow position.
Fuse the arm, forearm and elbow blobs.
Use the new blob and select all the points around the initial elbow position proposal within a 0.05 m square.
Calculate the centroid of the resulting blob, which will yield the elbow position.
end if

Algorithm 4 Elbow Correction

while the arm to elbow and elbow to forearm distances are incorrect (under an anthropometrically feasible threshold) **do**
if the elbow to forearm distance is above the threshold or the arm to elbow distance is under the threshold **then**
add lower points (belonging to the arm-forearm blob) to the blob used for the centroid calculation (steps 3 and 4 of the elbow calculation algorithm).
else if the elbow to forearm distance is under the threshold or the arm to elbow distance is above the threshold **then**
add upper points (belonging to the arm-forearm blob) to the blob used for the centroid calculation (steps 3 and 4 of the elbow calculation algorithm).
end if
end while

Hand Calculation and Correction

Due to the size of the hand blob, its labelling was noisy and consequently it was often not found or misplaced. This happened specially in situations were the hand was closed or sideways. So in order to correct erroneous hand positions an evaluation followed by a calculation was enforced (Figure 3.25). The evaluation (Algorithm 5) helped to remove hand positions that were not valid in relation to previous joints (in the kinematic chain), such as the elbow and the forearm. The calculation (Algorithm 6) provided a joint position proposal based on kinematic relationships when the evaluation considered the initial proposal not valid.

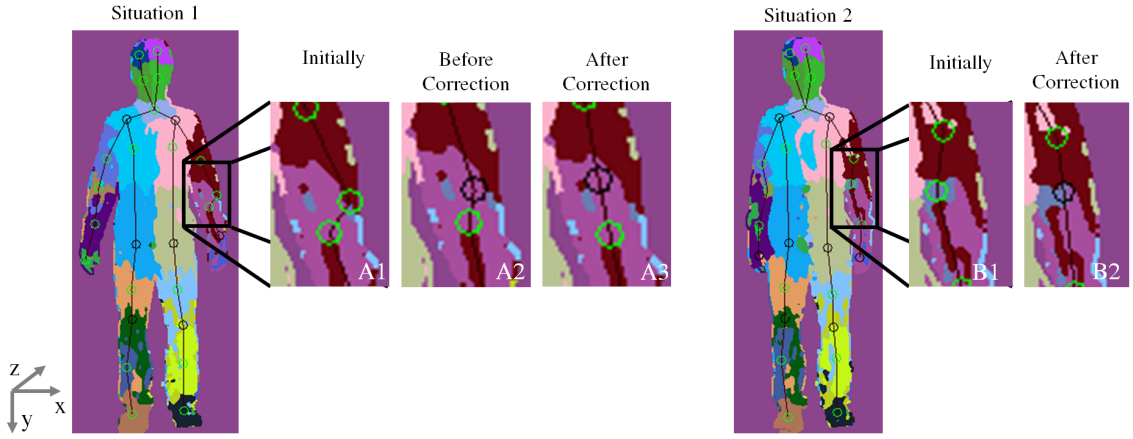

Figure 3.24: (Situation 1) Elbow position correction when the elbow blob is missing or it is too small to be considered valid. (A1) Elbow position returned by the initial algorithm, (A2) after the implementation of the proposed algorithm and (A3) after the correction. (Situation 2) Elbow position correction when the elbow blob is found. (B1) Elbow position given as the centroid of the elbow blob (initial algorithm) and (B2) elbow position after the correction enforcement. Inferred joints are presented has a black circle and not inferred joints as a green circle. The world coordinate system is presented in the lower left side of the figure. As can be observed, after the correction, the estimated elbow position occupies a centred position has would be expected.

Algorithm 5 Hand Evaluation

if Criteria 1: the forearm to hand distance is above or under a specified anthropometrically valid threshold OR Criteria 2: given a hand candidate (obtained by adding the elbow-forearm vector to the forearm position), the distance between the hand candidate and the hand proposal is bigger than 0.20 m **then**
▷ *Criteria 2 helps to remove hand proposals that have valid distances to the forearm but are not in a valid position (along the elbow-forearm vector).*
return hand position proposal not accepted
end if

Algorithm 6 Hand Correction

Fuse the forearm blob and the hand blob (if the hand position was rejected based on criteria 2, only the forearm blob is considered).
Remove all the points of the new blob above the forearm center, in order to avoid that the point with the maximum distance is found above the forearm center.
Get the point with the maximum distance from the forearm center within the new forearm blob, $(X_{max}, Y_{max}, Z_{max})$.
Select all the points around the found point within a 0.10 m square.
Calculate the centroid of the resulting blob, which will yield the hand position.

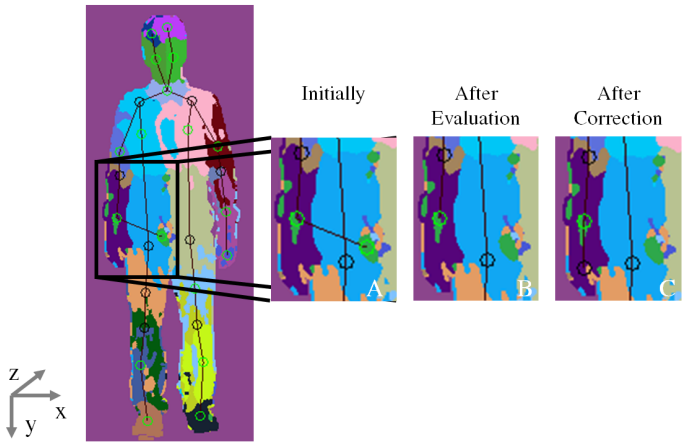

Figure 3.25: (A) Hand position before the evaluation, (B) after the evaluation and (C) after the correction. Inferred joints are presented has a black circle and not inferred joints as a green circle. The world coordinate system is presented in the lower left side of the figure. As can be observed in the left image, due to an incorrect labelling, the right hand proposal is not valid and so it is not accepted by the evaluation, being posteriorly correctly calculated by the proposed algorithm.

Hip Calculation

Since it is more clinically relevant to determine the hip bone position instead of the center of the hip blob (that would return a position around the belly button), only the lower points of the hip blob were considered for the hip joint calculation.

Algorithm 7 Hip Calculation

Get the lowest point of the hip blob $(X_{max}, Y_{max}, Z_{max})$.
Apply a passthrough filter, on the hip blob, to remove all the points with Y dimension outside Ymax-threshold.
Calculate the centroid of the resulting blob, which will yield the hip position.

Regarding the hips position calculation the previous implemented algorithm was stable and so was not modified (Algorithm 7). However, two thresholds were evaluated: 0.10 m (the threshold proposed by the previous implementation) and 0.30 m. Figure 3.26 presents the norm of the vector that goes from the left to the right hip joint during a sequence movement in which the person legs remains static. It would be expected that the distance should remain very similar through the entire sequence. As can be observed the threshold of 0.30 m provides more stable results and so it was chosen one. The threshold of 0.30 m was tested since, according to the evaluated images, was the one that, visually, estimated the hip position near the expected region.

Also, as can be observed in Figure 3.27, when a threshold of 0.10 m was enforced the proposed hip position was not reliable.

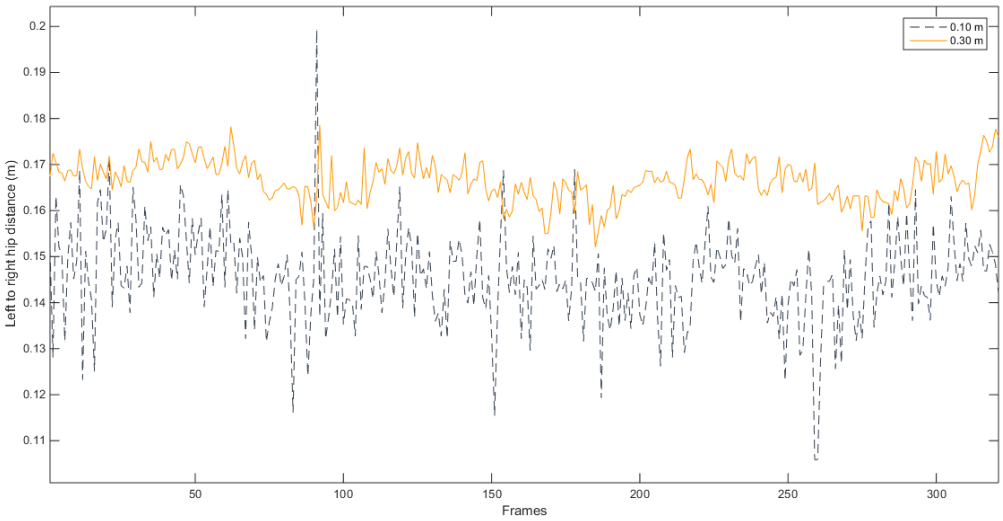

Figure 3.26: Left to right hip distance variation during a sequence movement in which the person legs remain static, using a threshold of 0.10 m (blue) or 0.30 m (yellow) for the hip position calculation.

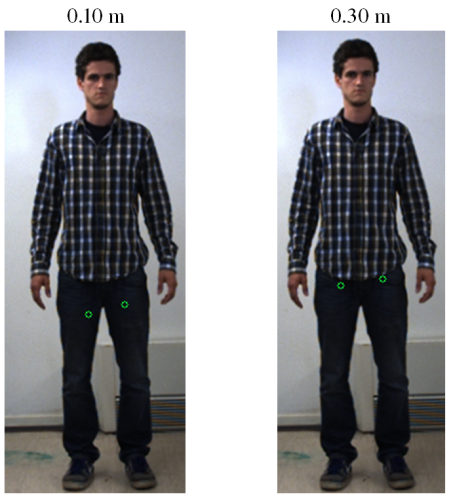

Figure 3.27: Hip joint position when a threshold of 0.10 m or 0.30 m is used for the position calculation. The returned hip position is marked by the green circles.

Thigh Correction

The thigh correction was introduced when the hip to thigh distance was above (1) or under (2) an anthropometrically valid threshold (Algorithm 8). This correction helped to prevent situations (among others) in which the left and right thighs were not parallel in the Y dimension in a standing position with static legs, due to an incorrect labelling, Figure 3.28A.

Algorithm 8 Thigh Correction

Set the limit as the double (1) (or half (2)) of the hip – thigh standard distance.
Remove all the points of the thigh blob that are above (1) (or under (2)) the limit.
Calculate the centroid of the resulting blob, which will yield the thigh position.

Knee Correction

Like the hand or the elbow blobs, due to the small size of the knee blob, its position was often misplaced. Or, in other situations, its location was often noisy and not accurate such as in the case presented in Figure 3.28B. Therefore a correction algorithm was implemented (Algorithm 9).

Algorithm 9 Knee Correction

if the thigh to knee distance is above or under an anthropometrically valid threshold **then**
Given the thigh blob remove all the points above the thigh center and below the knee candidate (the knee candidate is obtained by adding the thigh center to the thigh to knee standard distance).
Get the point with the maximum distance $(X_{max}, Y_{max}, Z_{max})$ from the thigh center within the blob returned by (1).
Fuse the thigh, knee and leg blobs.
Using the entire leg blob select all the points around $(X_{max}, Y_{max}, Z_{max})$ within square of 0.05 m.
Calculate the centroid of the resulting blob, which will yield the knee position.
end if

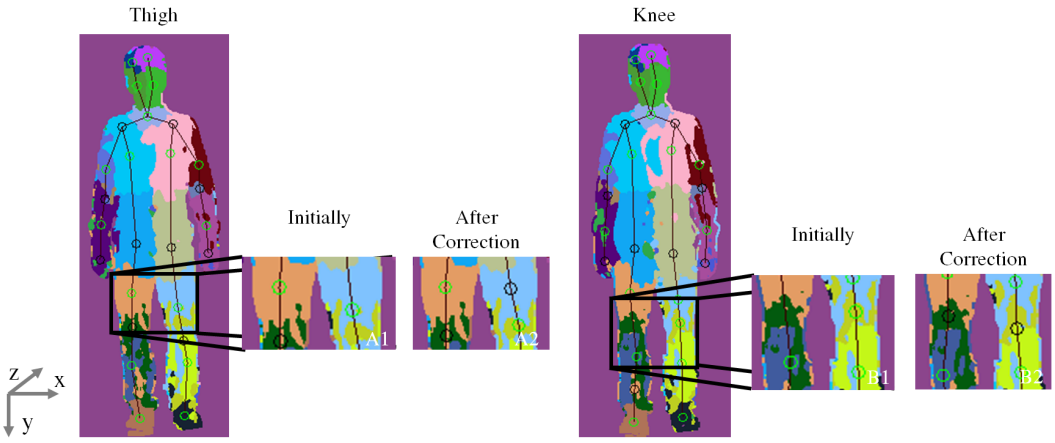

Figure 3.28: (A) Thigh position before (A1) and after (A2) the correction. (B) Knee position before (B1) and after (B2) the knee position correction. Inferred joints are presented has a black circle and not inferred joints as a green circle. The world coordinate system is presented in the lower left side of the figure. As can be observed, after the correction, the thighs and knees position are parallel to each other, like would be the expected in a standing position.

Leg Correction

The leg correction was enforced when, due to an incorrect labelling, the leg position was misplaced. Two situations were covered (Algorithm 10), one in which the foot blob was placed correctly (situation 1) and the other in which it was not (situation 2). When the leg position was found to be invalid its position was recovered based on the knee and foot positions, if the foot blob was

placed correctly (situation 1) or based on the knee and leg positions if the foot blob was not valid (situation 2). The leg position was considered not valid if its distance to its parent joint (knee) was above a anthropometric valid length. Figure 3.29 illustrates the two mentioned situations and the corrected result.

Algorithm 10 Leg Correction

if the knee to foot distance is valid given anthropometrically valid distance thresholds (the foot blob is valid) **then**
  Calculate a leg candidate position as the mean point of the knee to foot vector.
  if the leg proposal is located above a distance of 0.10 m of the leg candidate **then**
    the leg proposal is replaced by the leg candidate.
  else
    the leg proposal remains unchanged.
  end if
else if the knee to foot distance is not valid given anthropometrically valid distance thresholds (the foot blob is not valid) **then**
  Get the point with the maximum distance $(X_{max}, Y_{max}, Z_{max})$ from the knee within the leg blob or knee blob (if the leg blob is not valid).
  Select all the points around the found point within a 0.10 m square.
  Calculate the centroid of the resulting blob, which will yield the foot position.
  Given the foot and the knee position calculate the leg position as stated in case 1.
end if

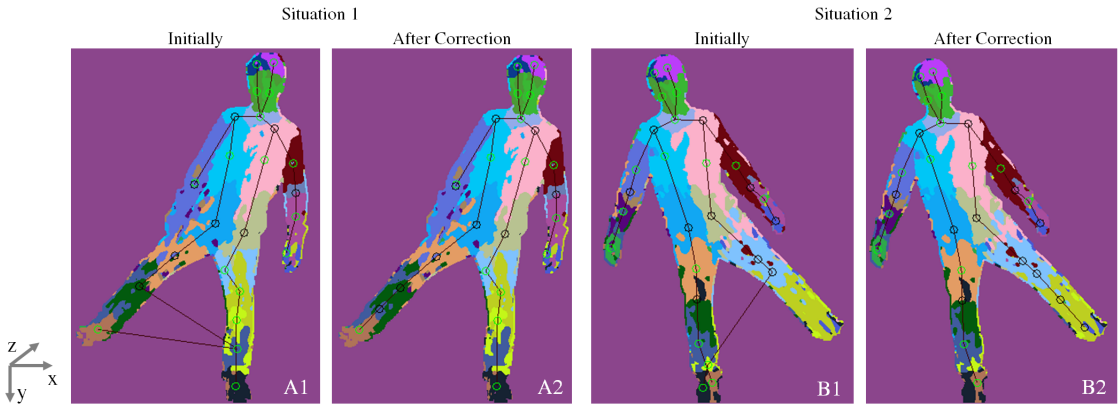

Figure 3.29: (A) Leg position before (A1) and after (A2) the correction for the situation 1. (B) Leg position before (B1) and after (B2) the correction for the situation 2. Inferred joints are presented has a black circle and not inferred joints as a green circle. The world coordinate system is presented in the lower left side of the figure. As can be observed after the correction the misplaced leg position was corrected, for both the situations.

3.3.4 Joints Position Tracking

Given the absence of temporal constraints, results can be prone to instability and jitter. For this reason, to improve the performance a Kalman filter was applied. Kalman filter was first introduced in 1960 [128] and has since been used in a wide variety of applications, including human motion

tracking [129]. The basic idea stated by the Kalman filter is that by observing a set of measurements of the system and considering some assumptions, a model of the system can be built that maximizes the posteriori probability of those previous measurements. Also, the maximization of the posteriori probability can be done without storing many previous measurements. This means that only the model for the next iteration is kept, after iteratively updating the model of the system's state. By doing it so, the computational expense of the described method is considerably decreased [96]. Nevertheless three assumptions need to be considered when using the Kalman filter:

1. the modelled system is linear,
2. the measurements are subjected to white noise,
3. the noise to which the measurements are subject is Gaussian.

The Kalman filter was applied to all the 27 retrieved joints, independently. The state vector, X , was considered to be the true 3D coordinates of the joints and their respective velocities, denoted as $x, y, z, \dot{x}, \dot{y}, \dot{z}$ (without the discrete time subscript t). Using the adopted coordinate system (in meters), at time t the state vector is:

$$X_t = \begin{pmatrix} x & y & z & \dot{x} & \dot{y} & \dot{z} \end{pmatrix}^T \tag{3.17}$$

As stated by the process model, the state at time t evolved from the prior state at time $t-1$ according to:

$$X_t = AX_{t-1} + w_{t-1} \Leftrightarrow \begin{pmatrix} x_t \\ y_t \\ z_t \\ \dot{x}_t \\ \dot{y}_t \\ \dot{z}_t \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & \triangle t & 0 & 0 \\ 0 & 1 & 0 & 0 & \triangle t & 0 \\ 0 & 0 & 1 & 0 & 0 & \triangle t \end{pmatrix} \begin{pmatrix} x_{t-1} \\ y_{t-1} \\ z_{t-1} \\ \dot{x}_{t-1} \\ \dot{y}_{t-1} \\ \dot{z}_{t-1} \end{pmatrix} + w_{t-1} \tag{3.18}$$

where A is the state transition matrix and w represents the normal distributed process noise with covariance Q. $\triangle t$ is the time step in seconds that was updated according to the time stamp of the acquired image frames.

The measurement model, that is composed by the 3D coordinates of each joint, relates the current state to the measurement Z with the matrix H. v is the normal distributed measurement noise with covariance R:

$$Z_t = HX_t + v_t \Leftrightarrow \begin{pmatrix} x_t \\ y_t \\ z_t \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_t \\ y_t \\ z_t \\ \dot{x}_t \\ \dot{y}_t \\ \dot{z}_t \end{pmatrix} + v_t \tag{3.19}$$

The used Kalman filter for the described model is detailed in the following steps:

1. *Initialization step:* input prior estimates $\hat{X}_0^-$ (state vector), P_0^- (initial covariance)
2. *Measurement update step:*

   (a) Compute the Kalman gain (K_t):

$$K_t = P_t^- H^T (H P_t^- H^T + R)^{-1} \tag{3.20}$$

   (b) Update state estimate:

$$\hat{X}_t = \hat{X}_t^- + K_t (Z_t - H\hat{X}_t^-) \tag{3.21}$$

   (c) Update the covariance:

$$P_t = (I - K_t H) P_t^- \tag{3.22}$$

3. *Time update step:*

   (a) Project the state ahead:

$$\hat{X_t^-} = A\hat{X}_{t-1}^- \tag{3.23}$$

   (b) Project the covariance ahead:

$$P_t^- = A P_{t-1} A^T + Q \tag{3.24}$$

In order to avoid feeding the filter with incorrect measurements, an evaluation function was developed. This function evaluates all the joints in relation to their parents joints and discards those that are considered to be invalid. The validation is based on the anthropometric distance between joints already described in the previous section. When a joint is considered to be invalid all the ones that follow on the kinematic chain are also set to invalid. When a joint is considered to be invalid the Kalman filter is updated by the prediction rather than the measurement, Figure 3.30.

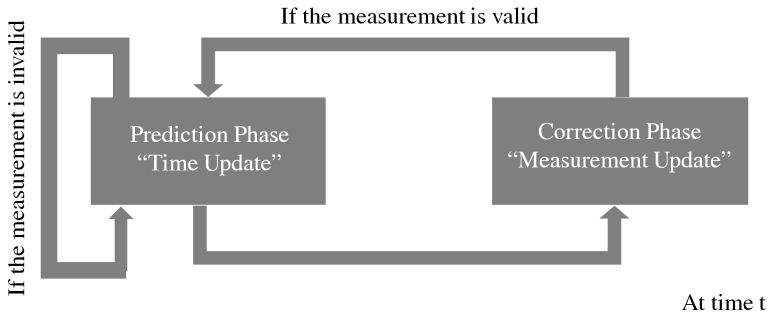

Figure 3.30: Block diagram describing the adopted Kalman filtering methodology. (Adapted from [96].)

The filter was initialized by setting $\hat{X}_0^-, P_0^-$ with the initial estimates of the skeleton joint positions with zero initial velocities. To initialize the error covariance of the state vector three types of skeleton joints were considered. The upper (with the exception of the hands) and the lower members were considered to be dynamic joints. According to the performed exercises, the dynamic joints were the ones where its position varied the most during the course of the exercises. The hands were considered to be a special case of the dynamic joints since they can move faster than the other joints. The torso and head joints were considered to be static joints. These joints were the ones which position remained the same (or almost the same) through the entire duration of the exercises. For this reason a higher value was attributed to the error covariance of the state vector for the hands, an intermediate for the dynamic joints and a lower one to the static joints. The error covariance of the state vector attributed to the three mentioned types of joints was a diagonal matrix with all entries equal to 0.0001, 0.000017, 0.0000029 respectively for the hands, dynamic joints and static joints. The process noise covariance (Q) was considered to be a diagonal matrix with all entries equal to 0.005. The measurement noise covariance (R) was considered to be a diagonal matrix with all entries equal to 0.05. The mentioned values were determined empirically.

3.3.5 Range of Motion Calculation

From the medical perspective, the direct evaluation of the skeleton joints position contains little information. For this reason, it is more important to provide quantitative and ready to use evaluation data calculated from the obtained joints positions. As well, in a rehabilitation context many physical therapies are based on the repetition to achieve a range of motion or control over a specific muscle group [58]. For this reason, the proposed rehabilitation exercises were evaluated by considering a set of quantitative measures, described in Figure 3.31:

1. **Arm abduction and adduction:** The first exercise was evaluated based on the shoulder angle. The shoulder angle was calculated as the angle between the shoulder to elbow vector and the neck to hip center vector. The hip center is not directly provided by the used skeleton model. For this reason, it was calculated as the medium point of the left hip to right hip vector [9]. During the arm abduction the arm would move from zero degrees to a value bigger than 90 degrees and then back to zero degrees, for each iteration. During the performance of this exercise the subject was instructed to performed the movement simultaneously for both arms.

2. **Hip abduction and adduction:** The second exercise was evaluated based on the hip angle. The hip angle was calculated as the angle between the neck to hip center vector (described in the previous item) and the hip to knee angle [9]. During the hip abduction, the abducting leg would move from zero degrees to a value beyond 45 degrees for each iteration. This means that the hip angle changes from zero to 45 (or bigger) and then back to 0 for each iteration and for each leg [8].

3. **Toe touch:** The third exercise was evaluated based on the hand to foot distance and the knee angle. As proposed in [8], to accommodate the size difference between different patients, the distance was normalized to the maximum distance (taken as the initial one). The boundary values should range between 1 in the beginning of the exercise and 0 at the end and in an ideal iteration (where the hands are able to reach the feet). Nevertheless, as this exercise is of considerable difficulty a minimum boundary should be customized according to each patient's capabilities [8]. The knee angle was calculated as the angle between the knee to hip vector and the knee to foot vector, for both the left and right knee. In order to ensure that the patients' legs remain straight during the entire exercise the knee angle should be always around 180°.

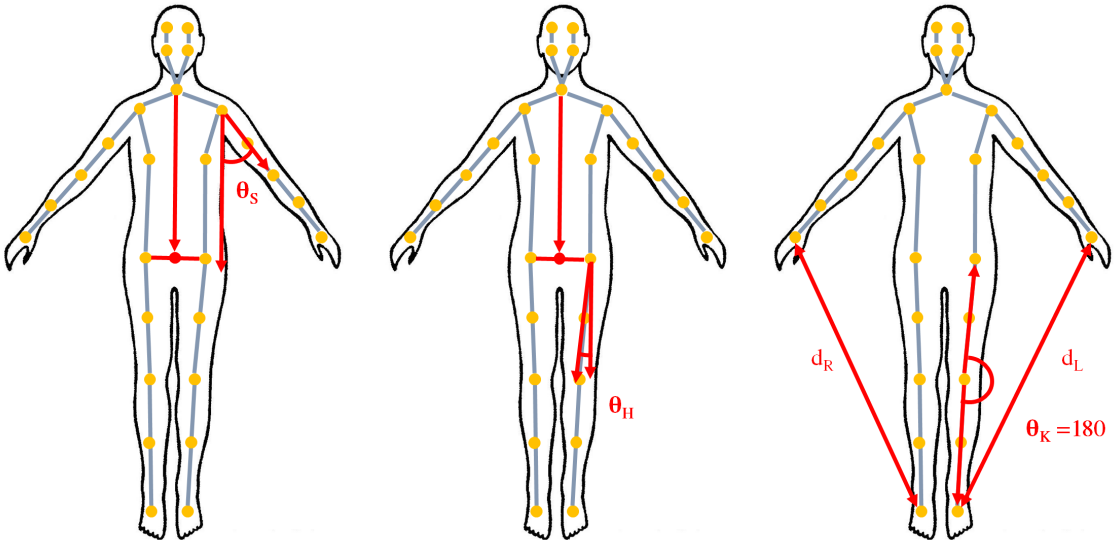

Figure 3.31: Measurement and calculation of the proposed quantitative evaluation of each rehabilitation exercise. For the shoulder (θ_S) and the hip (θ_H) angle the calculation is described for the left side. An analogous technique was used for the right side. The normalized hand to foot distance (d_R and d_L, for the left and right side respectively) was used to evaluate the performance of the toe touch exercise. Also, during this exercise the legs should remain extended and so the knee angle (θ_K) should remain equal to 180.

The aforementioned angles were calculated according to the following equation:

$$\theta = \arccos\left(\frac{a \times b}{\| a \| \cdot \| b \|}\right) = \arccos\left(\frac{x_a x_b + y_a y_b + z_a z_b}{\left(\sqrt{x_a^2 + y_a^2 + z_a^2}\right) \cdot \left(\sqrt{x_b^2 + y_b^2 + z_b^2}\right)}\right) \tag{3.25}$$

where $a = (x_a, y_a, z_a)$ and $b = (x_b, y_b, z_b)$ are the two vectors that form the angle of interest. For the shoulder angle calculation, a is the neck to hip center vector and b is the shoulder to elbow vector. For the hip angle calculation, a is the neck to hip center vector and b is the hip to knee vector. For the knee angle, a is the knee to hip vector and b is the knee to foot vector.

3.3.6 System Validation

The accuracy of the developed skeleton tracking system was evaluated using as ground-truth a marker based system. In order to record motions from both systems the stereo camera was mounted inside a motion capture laboratory. As shown in Figure 3.32 the stereo camera was placed at approximate 3 m from the subject. For the case of the optical sensor, the cameras cover the entire capture volume from multiple views.

Figure 3.32: Motion capture laboratory setup.

For the acquisition of the stereo images, the protocol described in Appendix A was followed. Given, the differences in the acquisition environment, the proposed pipeline for the segmentation (Figure 3.10) was unable to properly segment the subject. For that, a new segmentation pipeline was envisioned an is described in Appendix C. The methodology followed for the stages of point cloud generation and denoising were the same as the ones already described.

Regarding the marker based system, three-dimensional kinematics were monitored at 60 Hz using a 12 camera motion capture system (Qualisys AB, Gothenburg, Sweden) with an acquisition software (Qualisys Track Manager, Qualisys AB, Gothenburg, Sweden). This system tracks the spatial trajectories of the reflective markers on the subjects. The markers were placed in order to mimic the positions of the joints in the skeleton model used by the developed system (Figure 3.33A). For that, twenty seven reflective markers were placed on frontal bone of Cranium above the eyes (Face top), Zygomatic bone (Face bottom), Manubrium of Sternum (Neck), Acromion (Shoulder), the border between the 4th rib and connected costal cartilage (Chest), middle of lateral side of Humerus (Arm), lateral epicondyle of Humerus (Elbow), lateral side of Radius (50%) (Forearm), middle of the third Metacarpal (Hand), superior Ramos of Pubis (Hips), middle of frontal side of Femur (Thigh), Patella (Knee), middle of Tibia (Leg) and middle of second Metatarsal (Foot), for both the left and right sides [130] (Figure 3.33B-C).

Two healthy subjects, a male and a female (aged 23 and 22, respectively) were instructed to perform the exercises described in Section 3.1. For each subject and each exercise, three trials

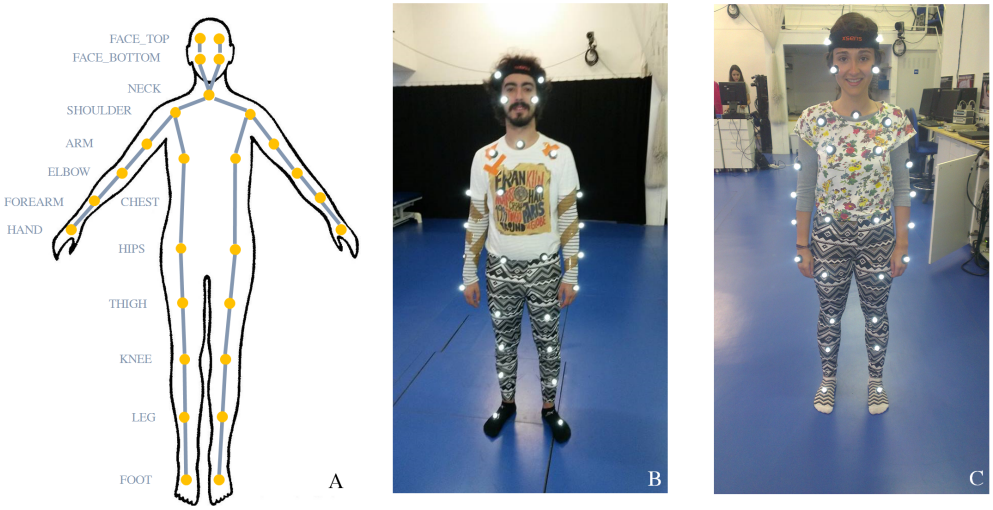

Figure 3.33: (A) Joints positions in the skeleton model used by the developed skeleton tracking system. Placement of the markers for both the (B) male and (C) female subjects.

were performed. In order to allow the synchronization of both systems (since they were independent manually activated and had different acquisition rates), the subject's were told to perform a clapperboard movement with both hands prior to each exercise. The mentioned movement marked the beggining of the time overlay. The end of the time overlay was considered to be the moment in which the subjects returned to the initial position.

Given the differences in the world coordinate systems of both systems the comparisons were performed based on the range of motion of the main articulations involved in the proposed exercises. For the calculation of the range of motion, for both the marker and the markerless systems, the conventions described in Sub-Section 3.3.5 were followed.

Based on the data provided by the marker based system, the ground-truth range of motion for each of the discrete timestamps of the markerless system was calculated based on a spline interpolation. When using this method the interpolated value at a query point is based on a cubic interpolation of the values at neighbouring grid points in each respective dimension. The analysis was performed using Matlab R2014b.

3.4 Summary

In this Chapter, the implemented methodology to perform skeleton tracking was proposed. The developed system had as input 3D information obtained by using a stereo camera. For this two distinct but yet connected stages were developed.

Initially, a 3D human body and its surrounding environment were acquired by a stereo camera. In order to improve the quality of the acquired 3D information a segmentation followed by a filtering and plane fitting stage were applied. The segmentation step took advantage of both the colour and depth information and improved the silhouette of the human body. Then, in order to

remove the noise and smooth the obtained 3D human body a bilateral filter was used. Due to the poor performance of the stereo matching algorithm in recovering the depth of untextured areas a plane fitting methodology was followed in order to improve the quality of the background.

Finally, the enhanced point cloud was used as input for the skeleton tracking system. The proposed system uses a pixel-wise body part labelling to construct an intermediate body part representation from which valid skeletons are retrieved. Nevertheless, a correction stage based on kinematic relationships and anthropometrically feasible lengths between joints was implemented to improve the quality of the recovered skeletons. Furthermore, in order to take advantage of temporal consistency and remove some of the jitter in the obtained skeleton joints, a Kalman filter tracking approach was developed.

Considering the final goal of using the developed system in a context of rehabilitation and according to the proposed rehabilitation exercises, clinically relevant information was extracted from the obtained skeleton data, considering a set of range of motion measures. Furthermore, ground-truth information using a marker based system was recorded to evaluate the performance of the developed system.

Chapter 4

Results and Discussion

In this chapter the performance of the methodologies described in Chapter 3 is evaluated and discussed. Furthermore, the main advantages and disadvantages of the chosen pipeline are presented, as well as how those choices influenced the subsequent stages.

As highlighted by the previous chapter, the present work was divided into two stages. The first, consisted in the acquisition of a 3D point cloud representing the appearance of the human body and the second where the 3D representation served as input for the skeleton tracking system.

The first stage of the described pipeline was tested on a Windows 64-bit Intel Core i5-3317U CPU at 1.70 GHz, with 8 GB of RAM computer system. The second stage was tested on a Windows 64-bit Intel i7-2600 CPU at 3.40 GHz, with 8 GB of RAM and a Nvidia GeForce GTX 650 GPU. Due to the computational weight of the pixel-wise body part labelling determination, this task of the skeleton tracking system was implemented on the GPU. This resulted in the need for the use of two computers.

To evaluate the performance of the developed system, three image sequences were acquired in which a single male individual performed a set of three rehabilitation exercises (see Section 3.1 for further details):

- **Sequence 1:** Arm abduction and adduction, first in the coronal plane and them in the sagittal plane.
- **Sequence 2:** Hip abduction and adduction in the coronal plane with the knee extended.
- **Sequence 3:** Toe touch.

For each image sequence a total of 400 frames were collected. The image sequences were acquired with the stereo camera Bumblebee®2 from Point Grey. The camera was placed in order to allow full acquisition of the subjects body and ensuring that only the person was in the line of sight of the camera.

4.1 Human Body Reconstruction

The developed skeleton tracking system, described in Section 3.3, had as input a 3D representation of the human body, obtained using a stereo camera.

According to the preliminary assessment described in Section 3.2.1, the chosen stereo matching algorithm was the SGBM. Nevertheless, as described in literature [17], stereo matching algorithms have the disadvantage of performing poorly in untextured or dark surfaces. Since the goal was to obtain a 3D reconstructed human body the optimal conditions regarding the subject's clothing were tested. For that, several video sequences of subjects wearing textured and plain clothes were acquired and their respective 3D point clouds analysed. As presented in Figure 4.1, when the subject wears untextured clothes the raw points clouds are noisier. The mentioned noise is more noticeable on the irregularity of the reconstructed 3D point cloud of naked body parts (subject's arms in Figure 4.1B) or plain colour clothing (subject's torso in Figure 4.1B).

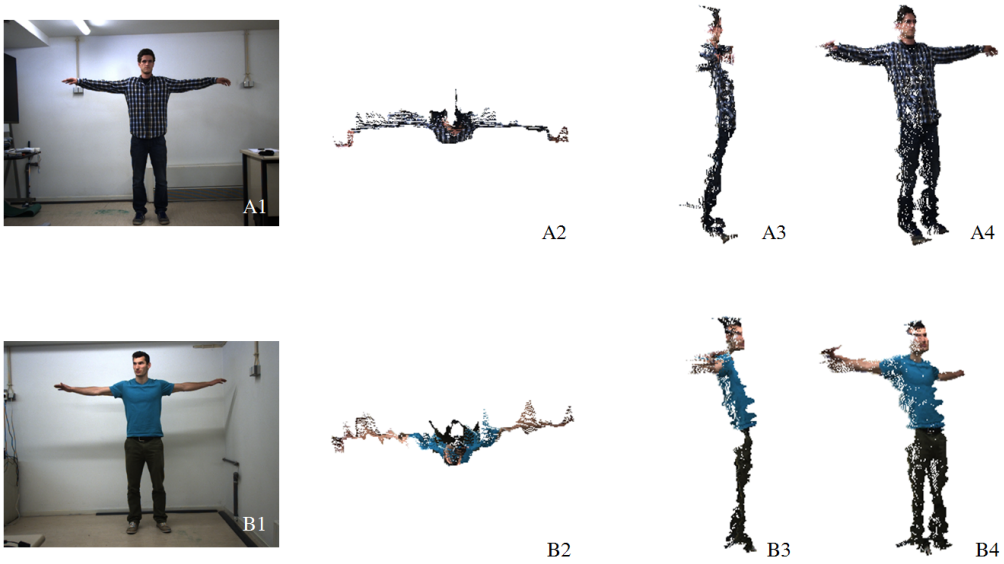

Figure 4.1: Comparison of the raw 3D point clouds obtained when the subject is wearing (A) textured and (B) untextured clothes. (A1,B1) Reference RGB Image. Raw point clouds in (A2,B2) top, (A3,B3) lateral and (A4,B4) diagonal views.

As previously mentioned in Section 3.2.2, due to the nature of the SGBM stereo matching algorithm, the contours of the foreground on the 3D raw point clouds were not clearly defined and so a segmentation pipeline, Figure 3.10, was developed in order to correct those situations. The results obtained with the proposed segmentation pipeline were visually assessed, Figure 4.2. Despite achieving satisfying results in most cases in rare situations some areas were not properly segmented. Nevertheless, given the overall evaluation of the proposed segmentation pipeline on the acquired images, the returned results were considered satisfactory.

In the absence of texture, stereo matching algorithms have some difficulty in recovering the depth information. Since the acquired images had as background a plane white wall the SGBM algorithm was unable to return a consistent disparity map. For the stated reasons, synthetic backgrounds were created following a plane fitting approach, described in Section 3.2.2. In comparison with the wall, the floor was more textured and so the depth recovery was better. Nevertheless, in order to smooth the floor the same plane fitting methodology was applied. The obtained plane

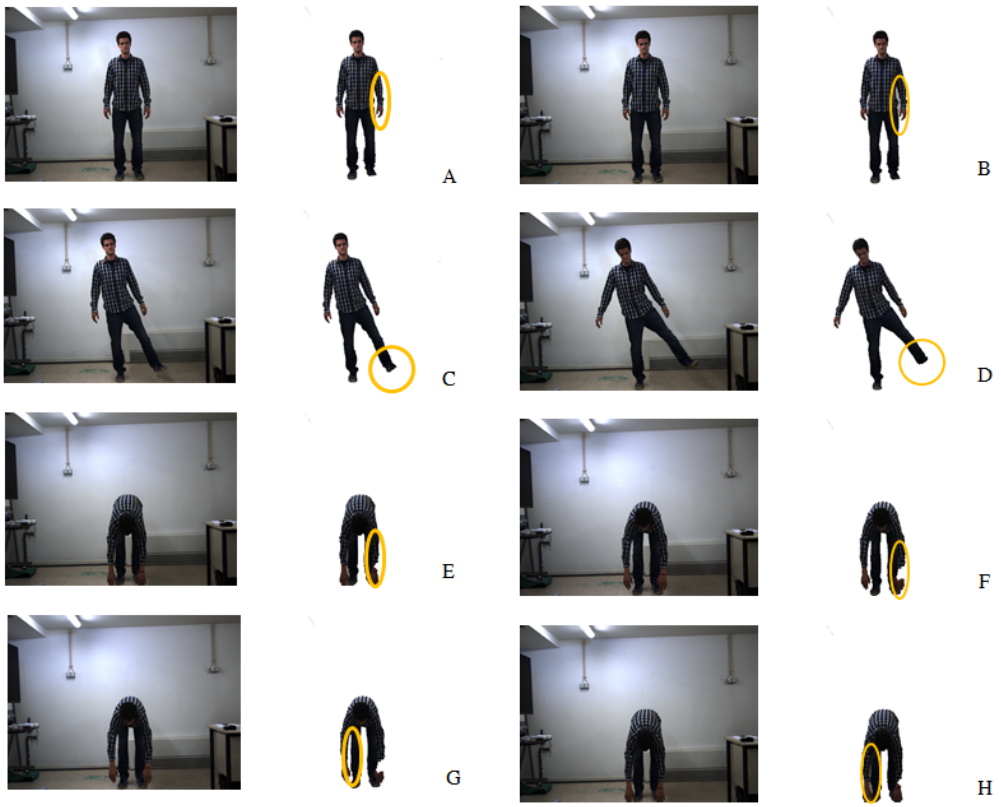

Figure 4.2: Comparison of similar frames in which the proposed segmentation pipeline performs well (first column) and in which it performs poorly (second column). The well segmented and miss segmented regions are highlighted by yellow circles. The original RGB images are presented in the left side for comparison. (D,F) Examples of over-segmentation and (B,H) segmentation by default are presented.

coefficients for the wall and floor planes are presented in Table 4.1. Besides solving the aforementioned problem of lack of texture, the proposed plane fitting approach in combination with the segmentation allowed the projection of all the objects that do not belonged to the human body (such as the table, presented in the right lower corner of the point clouds of Figures 4.3) to the wall plane. This simplified the determination of the subject's position in the subsequent skeleton tracking system.

Table 4.1: Plane coefficients returned by the RANSAC algorithm for the wall and the floor plane.

Plane coefficients	Wall Plane	Floor Plane
a	0,0652662	-0,331952
b	-0,0693572	-0,999442
c	0,995455	-0,00361078
d	-4,78058	1,04805

The impact of the proposed segmentation and denoising methodology in the enhancement of the final point clouds is presented in Figure 4.3. As shown, the proposed pipeline is able to improve the quality of the final point clouds.

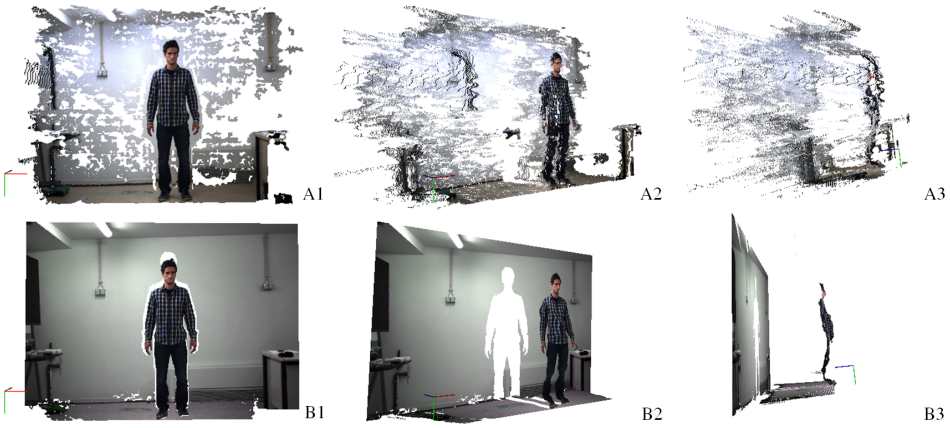

Figure 4.3: Obtained 3D point clouds (A) before and (B) after segmentation and denoising. (1) Frontal view, (2) diagonal view and (3) lateral view. The world coordinate system is presented for guidance: z-direction is given by the blue axis, the y-direction by the green axis and the x-direction by the red axis.

For each suggested rehabilitation exercise, two clouds are presented for comparison, Figure 4.4 to Figure 4.6. The proposed pipeline was able to reconstruct the human body not only in the coronal plane as well as in the saggital plane. Nevertheless, when the segmentation fails the quality of the resulting point cloud is degraded, Figure 4.6B. In agreement with the presented point clouds one can affirm that the proposed pipeline for human body reconstruction was able to return reliable results that can be used as input for the skeleton tracking system discussed in the following section.

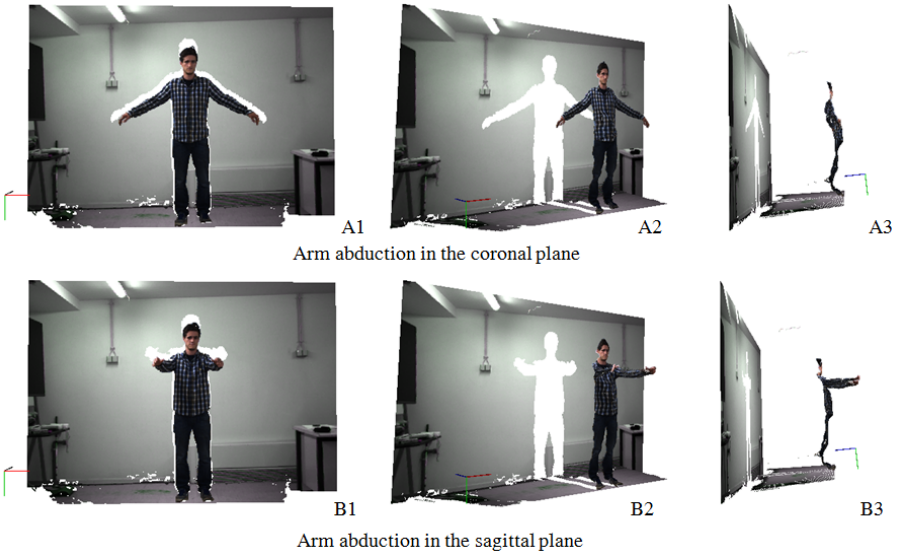

Figure 4.4: Obtained 3D point clouds after segmentation and denoising. (1) Frontal view, (2) diagonal view and (3) lateral view. The presented clouds are from the first sequence of rehabilitation exercises. The world coordinate system is presented for guidance: z-direction is given by the blue axis, the y-direction by the green axis and the x-direction by the red axis.

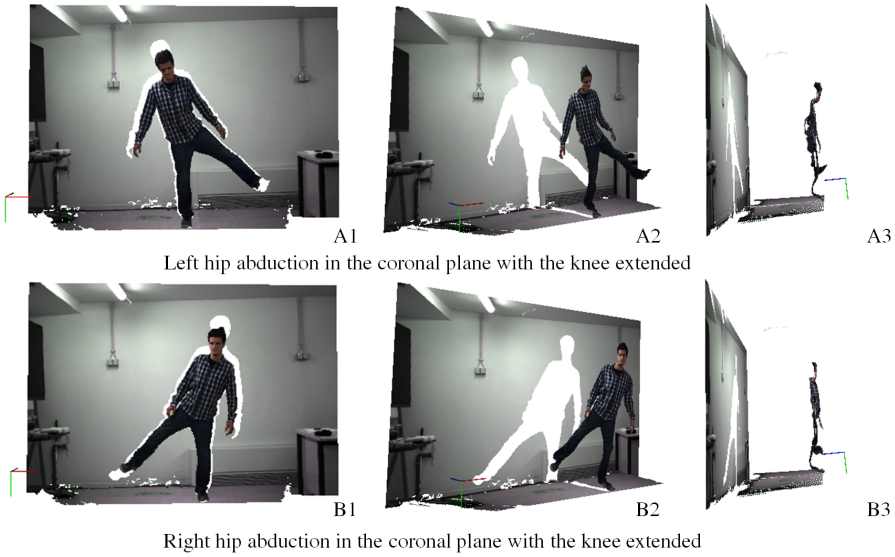

Figure 4.5: Obtained 3D point clouds after segmentation and denoising. (1) Frontal view, (2) diagonal view and (3) lateral view. The presented clouds are from the second sequence of rehabilitation exercises. The world coordinate system is presented for guidance: z-direction is given by the blue axis, the y-direction by the green axis and the x-direction by the red axis.

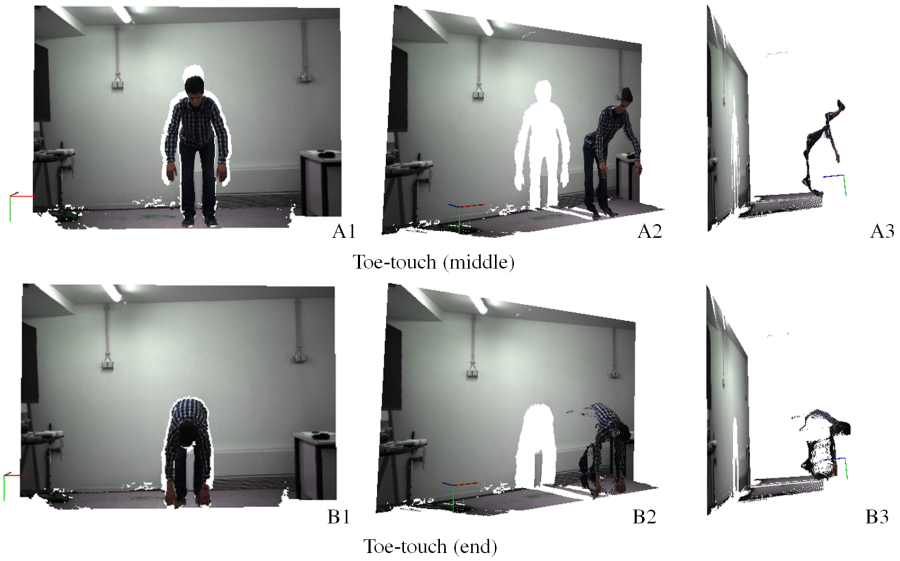

Figure 4.6: Obtained 3D point clouds after segmentation and denoising. (1) Frontal view, (2) diagonal view and (3) lateral view. The presented clouds are from the third sequence of movements. The world coordinate system is presented for guidance: z-direction is given by the blue axis, the y-direction by the green axis and the x-direction by the red axis. It is worth to note that in the fourth line a failed segmentation near the subject's right arm deteriorated the resulting point cloud.

4.2 Human Pose Estimation and Motion Tracking

At this point a 3D representation of the human body and its surrounding environment has been generated. The obtained 3D information was used as input for the skeleton tracking system. Due to the absence of annotated ground-truth data (for the initially obtained image sequences) the presented results are limited to the evaluation of the consistency in person and skeleton joints detection. The person detection consistency was assessed considering the person detection rate. The consistency in skeleton joints detection was assessed considering the kinematic and length relationships between the returned skeleton joints positions.

The adopted algorithm was inspired in the one of Shotton et al. [84]. The main resemblance is found in the use of an intermediate body part representation. The intermediate representation, from which the joints positions were predicted, was obtained trough pixel-wise body part labelling. The process of pixel labelling was accomplished through the use of an RDF classifier that was learned using the same depth features as the ones described in [84]. The output of this stage was a set of connected regions each one representing a body part proposal, Figure 4.7. As can be observed, the algorithm predicts multiple proposals per joints. Only the most confident proposal (represented by the larger area) was considered for the joints position calculation. From the presented labelled frames, a few failures were evident:

1. the system had some difficulties in correctly labelling the hands and the elbows when they are close to the torso, Figure 4.7A;
2. when the movements are performed outside the coronal plane, the labelling is deteriorated, Figure 4.7C;
3. as the movements begin to diverge from the ones presented in the training dataset, the labelling performance is degraded, Figure 4.7C;
4. the system in unable, in most cases, to correctly label the ground plane, Figure 4.7A-C,E.

As mentioned in [16], due to the support of the kinematic chain and the on-line learning the algorithm was able to deal with poses outside the training set. Nevertheless, one can hypothesize, that in those situations the labelling performance would be less reliable. The absence of the pose presented in sequence 3 on the training data supports the poor performance of the labelling process. As well, the inability of the system to correctly label the floor was due to the absence of the ground plane in the training data. Other factors may influence the labelling performance such as the camera angle position and the human body model used to generate the depth images used for training. The current training data was generated using a straight-on, chest-height camera angle with a single slim male model in an uncluttered environment. Other similar training datasets used for pixel-wise body part labelling, such as the one in [131], account for a far larger amount of variations, including several body models, from males to females, from child to adult considering height and weight variations and different camera positions and orientations.

Unless otherwise specified all the presented labelling results were obtained with the following training parameters: the RDF was trained using a single slim male MakeHuman model in an uncluttered environment; each forest was trained with 3 trees; each tree was trained to a depth of

20, with 80k images per tree.

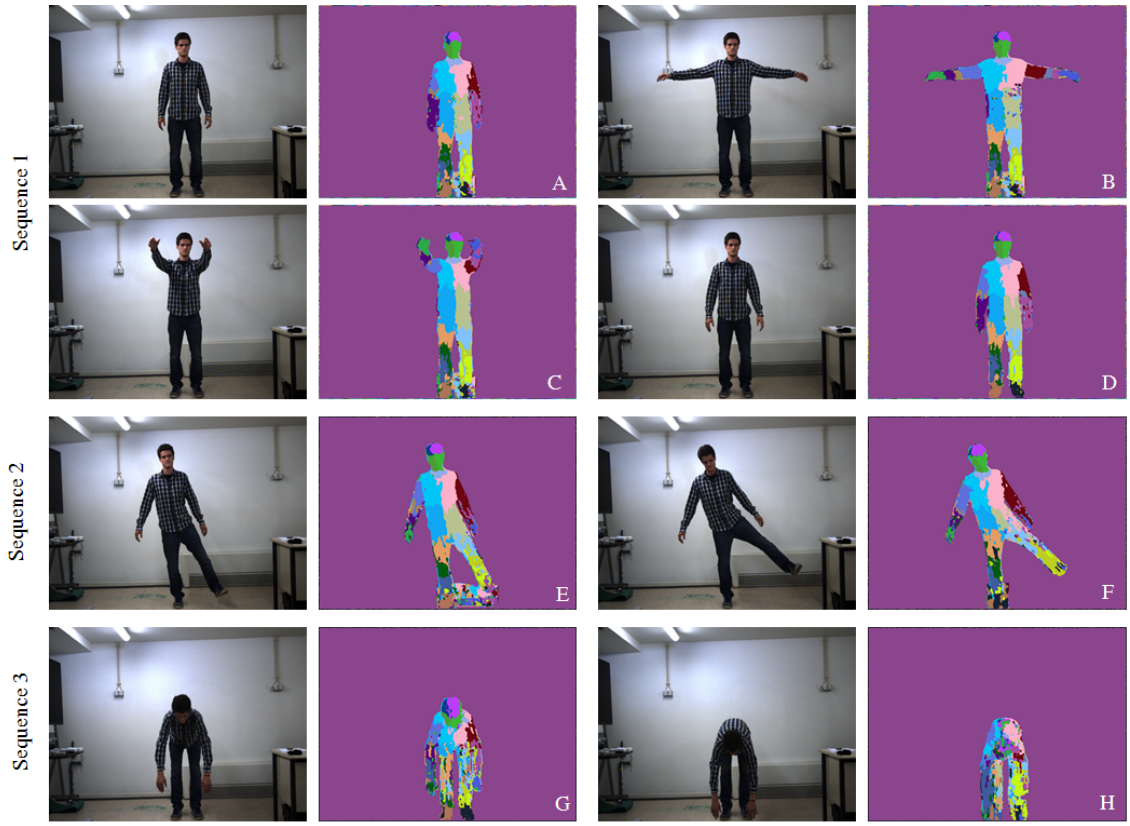

Figure 4.7: Output of the pixel-wise body part labelling for selected frames of the three exercise sequences. For each labelled frame, the original RGB reference image is presented on the right for comparison. The images are color coded to improve readability. Each color represents a label and consequently a body part proposal. (A) When the arms are close to the torso the system was unable to correctly label the hands and the elbows. (B) In opposition, when the subject assumes a T pose, the elbows and hands were correctly labelled. (C) The inability to correctly identify the ground plane is well noted by the noisier labelling around the feet. Nevertheless, in some situations (D), the system was able to correctly estimate the ground plane. (E) As the subject performs the hip abduction, the labelling was deteriorated, (F) being even unable to distinguish between the right and left foot when the leg reaches the maximum aperture. (G-H) When the position was very different from the ones presented in the training set the labelling outcome was not consistent.

The pixel-wise body part labelling was also evaluated based on the person detection rate, Figure 4.8. Considering all the images containing a person, the person detection rate defines the percentage of those images in which a person is detected. In general the person detection rate was above 70%. As expected, due to the poor labelling performance for the exercise sequence 3, the person detection rate is lower for this sequence.

The system's inability to correctly label the floor is demonstrated in Figure 4.9. This contributed to the use of the Ground Plane Detector (Section 3.3.2). The detector was able to estimate the floor plane and remove it. As well, an initial people detection was performed, in which the point cloud cluster where the subject is located was estimated. Only this point cloud cluster was passed to the labelling stage and so the person detection rate was improved, Figure 4.8.

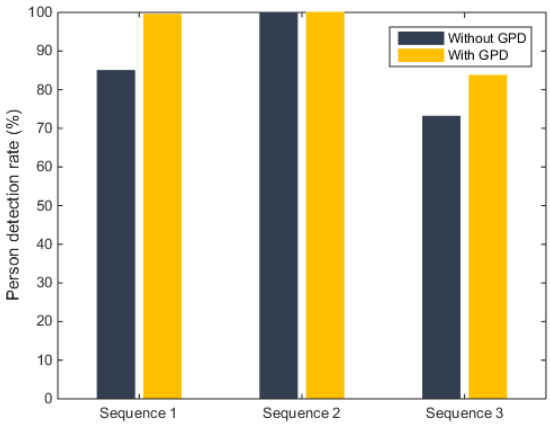

Figure 4.8: Person detection rate of each exercise sequence, when the subject's detection is aided by the Ground Plane Detector (GPD) (yellow) and when it is not (dark blue).

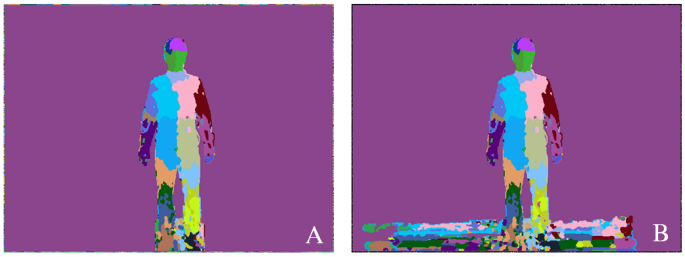

Figure 4.9: Comparison of the labelling outcome for the same frame when the subject's detection is aided by the Ground Plane Detector (A) and when it is not (B). The use of the Ground Plane Detector tried to overcome the system's inability to correctly label the ground plane. This was accomplished by removing the ground plane from the point cloud before passing it to the labelling step.

To remove the noise of the obtained raw point clouds a bilateral filter was applied. This filter allows image smoothing thanks to the domain component while still preserving the edges due to range component. Besides smoothing the raw point clouds, as shown in the preliminary results of Section 3.2.2, the use of the filter improved the labelling outcome, by reducing the number of mislabelled small patches (mainly noticeable on the torso region), Figure 4.10.

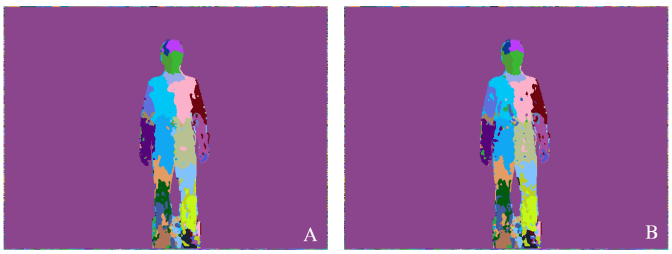

Figure 4.10: Comparison of the labelling outcome for the same frame (A) with and (B) without the use of the bilateral filter. As can be observed the use of the filter contributes to a smoother labelling result.

As referred in Section 3.3.1 the used trees for the body part labelling were provided by PCL and adapted to work with the data provided by the stereo camera through an up-scaling. Despite the presence of some evident problems, namely the difference between the evaluated poses and the ones present in the training, the available trees were not retrained. This was a result of the difficulty in obtaining real motion capture data, from a marker based system, such as the Vicon system, that suited the evaluated exercises. For this reason the main focus was on improving the joints positions returned from the labelling output. Given the instability of some of the returned joints, their position was reviewed and corrected (Section 3.3.3). These joints were the shoulders, elbows, hands, hips, thighs, knees and legs. The proposed correction algorithms were evaluated considering the percentage of invalid joints (I):

$$I_i = \frac{1}{F}\sum_{f=0}^{F} \delta_{inv}, i = 0, ..., 26 \tag{4.1}$$

where i represents each one of the 27 retrieved joints, F is the total number of frames and $\delta_{inv} = 1$, if the correspondent joint position is considered to be invalid. The joints validity was assessed according to the anthropometrically feasible lengths described in Section 3.3.3.

The percentage of invalid joints for the three evaluated sequences is presented in Figure 4.11. An overall decrease of invalid joints is visible after the correction implementation for all the sequence movements. This improvements sustain the viability of the implemented correction. Due to the increasing movement complexity from sequence 1 to sequence 3, an increasing in the percentage of invalid joints is also noteworthy. One can observe that the neck, the shoulders and the chest are the most stable joints. This is related to the fact that all the skeleton proposals are built from the neck, with the shoulders and the chest being the child and grandchild joints of the neck, respectively. Another considerable improvement is visible regarding the hands position, with an overall decrement of 52% to 13% considering both left and right hands.

The impact of the implemented correction stage on the overall consistency of the obtained skeletons is shown in Figure 4.12. As proved by the previous quantitative analysis, the implemented corrections improved the overall quality of the returned skeletons. The most remarkable improvements can be seen in the hands, elbows, hips and knee positions. Due to an incorrect labelling, the hands were often misplaced. By enforcing feasible lengths those misplacements were corrected, Figure 4.12A-B. Owing the small dimension of the elbow label its position was often not determined or in the cases in which it was determined it didn't occupied a central position. As shown, the implemented correction was able to correctly place the elbow even when its label was missing, Figure 4.12B, and when it position was not central, Figure 4.12C. This latter improvement was not contemplated in the previously described quantitative assessment (an elbow position can be at a feasible distance in relation to the arm and do not occupy a central position). During most part of the proposed rehabilitation exercises, the subject's legs remain static. This means that a certain level of parallelism between the corresponding left and right joints of the legs should be observed. By taking into account anthropometrically valid lengths between joints during the correction stage, the aforementioned parallelism was accomplished.

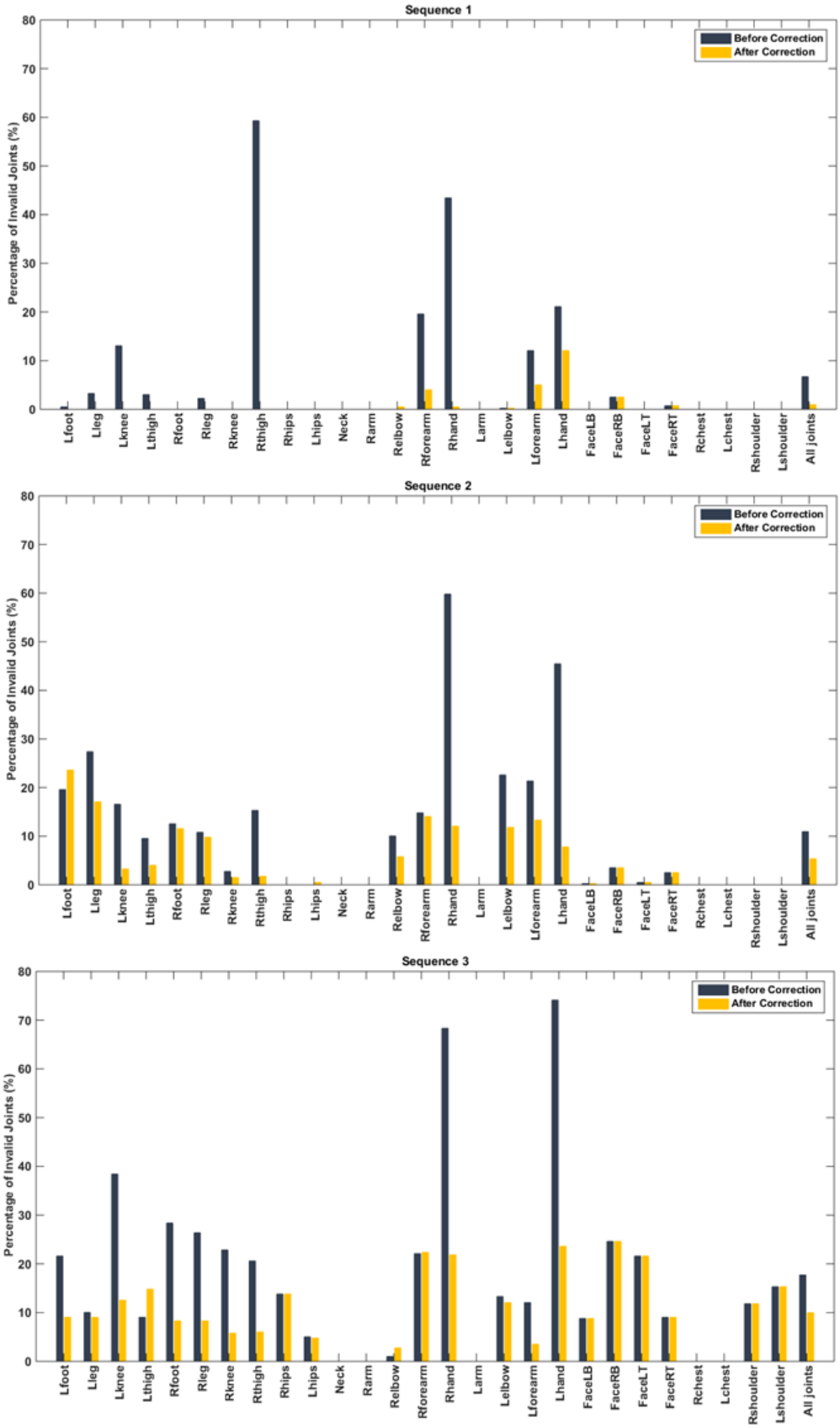

Figure 4.11: Percentage of invalid joints before (dark blue) and after (yellow) the implementation of the new correction algorithms for the all the image sequences. (Sequence 1) Arm abduction and adduction. (Sequence 2) Hip abduction and adduction. (Sequence 3) Toe touch.

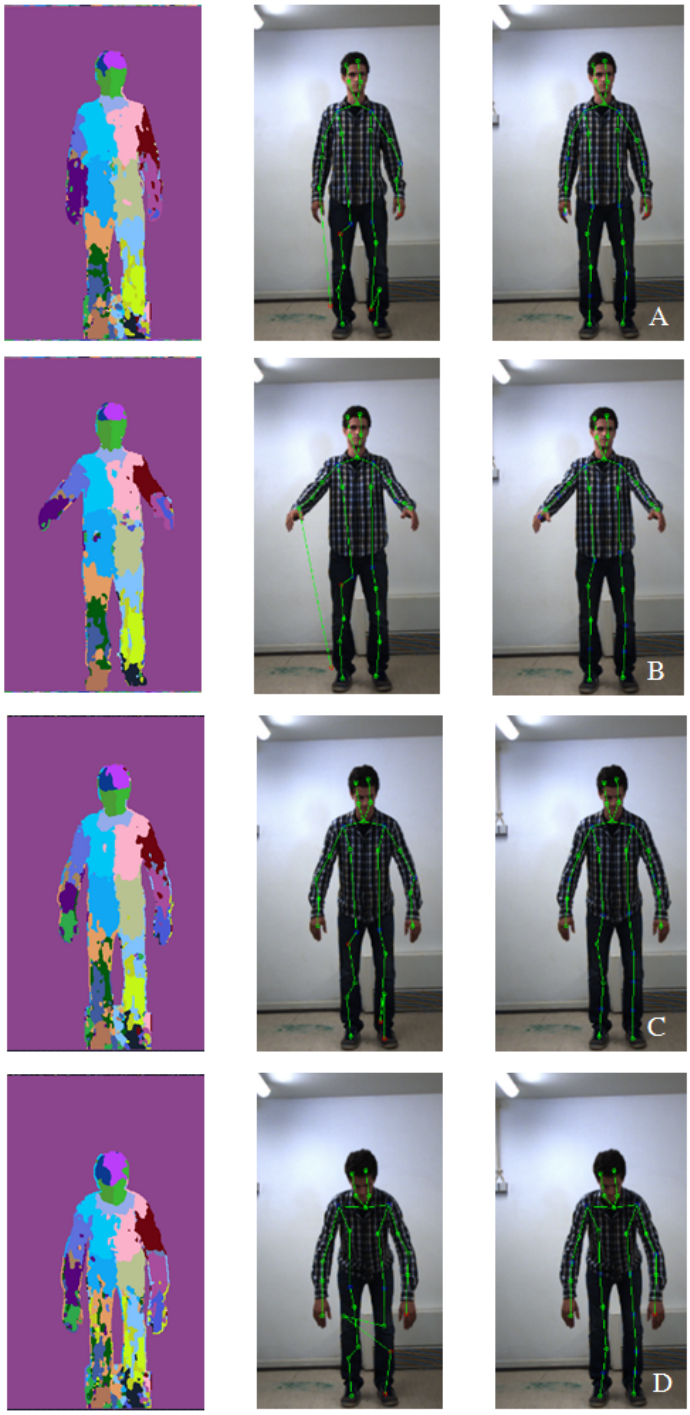

Figure 4.12: Impact of the implemented correction stage on the overall consistency of the obtained skeletons for selected frames. For that, the returned skeleton joint positions before (middle) and after (right) the implementation of the new correction algorithm are presented. For each frame, the correspondent labelled mapping is presented for comparison (left). Inferred joints are presented as a blue circle, not inferred joints as a green circle and invalid joints as a red circle. The returned skeletons are superimposed in the RGB images in order to improve the visual assessment. The images were cropped to improve visualization.

Despite the improvement provided by the implementation of the correction stage, the returned skeleton joints positions presented a considerable amount of jitter. As well, due to a bad labelling performance some of the obtained joints positions in individual frames were not accurate. In order to solve the aforementioned problems a Kalman filter was implemented, as described in Section 3.3.4.

Figure 4.13 presents the comparison between the x, y and z trajectories of the raw and Kalman filter estimate data for selected joints (the information regarding the remaining joints can be consulted in Appendix B). Since the first sequence returned the most stable results after the correction implementation, only that sequence was used to visually evaluate the use of the Kalman filter. For all the three spatial coordinates the output of the Kalman filter presented a much more smother estimate in comparison to the directly obtained raw data from the skeleton tracking system that tended to shake between time steps. The mentioned smoothness is also patent when the same evaluation is done for the sequences 2 and 3 (data not shown). As well, as presented in Figure 4.13A, the Kalman filter was able to accurately predict the joints position in the absence of measurement information (when the skeleton tracking system was not able to return a joint position, the x, y and z coordinates were marked as -1). Moreover, the implemented filter prevented and recovered big jumps in the 3D position of the joint in individual frames as can be visualized in Figure 4.13D.

To quantify the observed smoothness the following measure was introduced [132]: for each frame f the absolute position, $a_{f,j}$, of each joint j in the human skeleton was taken and the movement from the previous frame was measured. The smoothness measure (S) was then calculated as the average deviation of all joints J over all frames F:

$$S(x_{1:T}) = \frac{1}{FJ} \sum_{f=0}^{F} \sum_{j=0}^{J} \| a_{f,j} - a_{f-1,j} \| \tag{4.2}$$

The smoothness measure was calculated for the three sequences. The results for the raw data and the Kalman filter estimates are shown in Table 4.2. As would be expected, the Kalman filter estimates present a lower deviation between frames. As well, due to skeleton tracking system's inability to return stable results for the third sequence, the smoothness measure was considered unreliable for the raw data.

Table 4.2: Smoothness of raw data and the Kalman filter estimate measured by the average deviation of absolute joint positions between frames. Results are presented in meters. Low values indicate smooth trajectories.

Sequence	Raw Data	Kalman Filter Estimate
1	0.0181	0.0065
2	0.0393	0.0078
3	6.9330e+15	0.0058

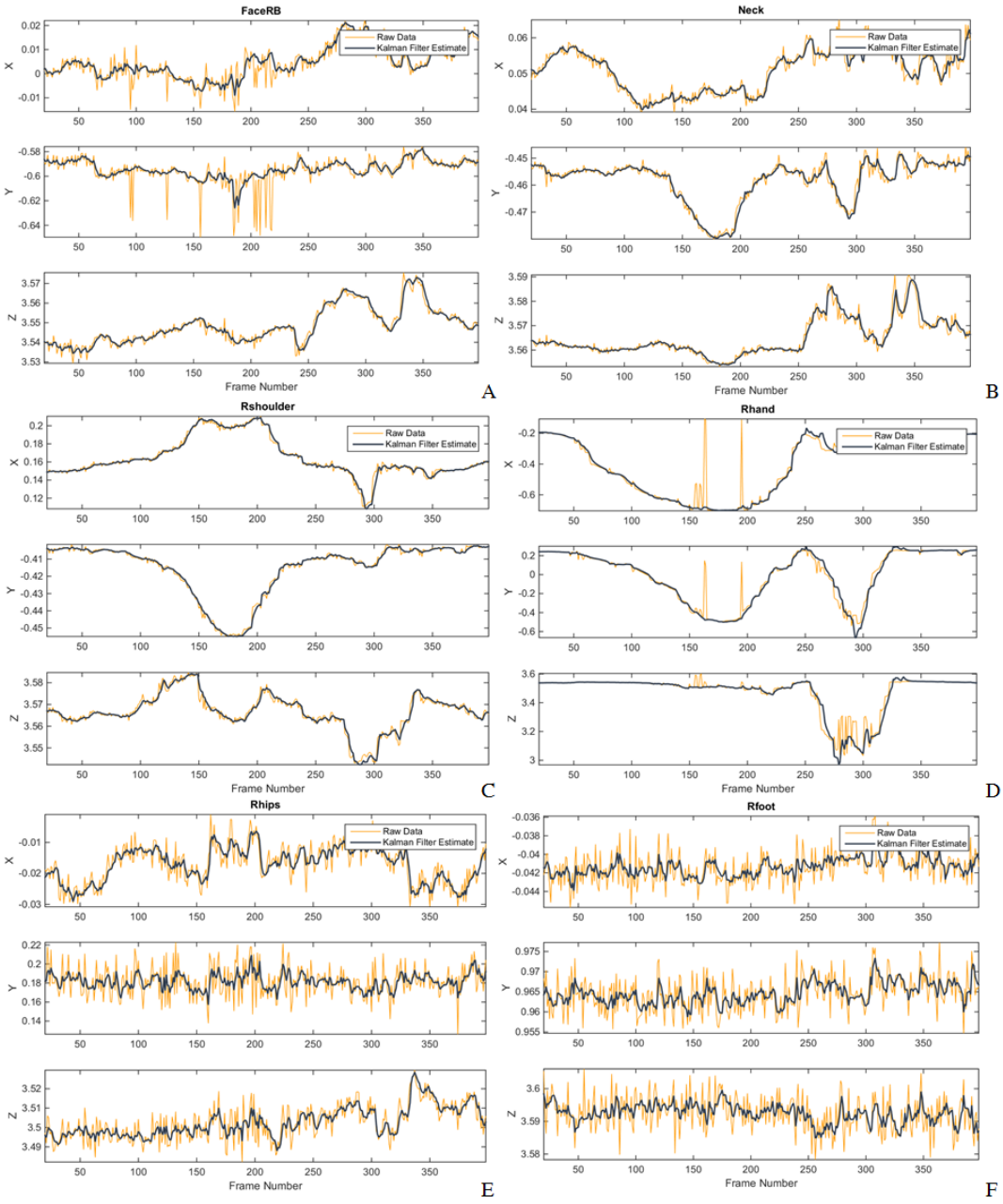

Figure 4.13: Comparison of the Kalman filter estimate (dark blue) and the raw data (yellow), in meters, for the x (top), y (middle) and z (bottom) trajectories. The same evaluation was performed for all the 27 joints. Only the (A) face right top, (B) neck, (C) right shoulder, (D) right hand, (E) right hip and (F) right foot trajectories are presented for comparison. The correspondent results for the remaining joints can be consulted in Appendix B. When the skeleton tracking system was not able to return a joint position, the x, y and z coordinates are marked as -1.

The performance of the proposed rehabilitation exercises was evaluated based on the output of the skeleton tracking system. For evaluation both the raw data and the kalman filter estimates (that consists on the final output of the skeleton tracking system) were considered. Results are presented in Figures 4.14 to 4.16.

As previously highlighted the system performance degrades with the increasing complexity of the evaluated exercises. Also, as expected, the kalman filter estimates present a smoother trajectory in comparison to the raw data.

The arm abduction and adduction was evaluated considering the shoulder angle (Figure 3.31). The movement was performed initially in the coronal plane and then in the saggital plane. The subjects were instructed to performed the arm abduction until achieving a shoulder angle of around 90º, maintaining that position for close to 5 seconds and then returning to the initial position. As shown in Figure 4.14 the skeleton tracking system was able to estimate the trend of the movement, achieving close to 90º during the plateau stage in both the coronal plane and the saggital plane.

The hip angle (Figure 3.31) was considered to evaluate the hip abduction and adduction. For this exercise, the subjects were told to maintain the knee extended and perform the hip abduction until reaching its maximum aperture trying to contain the movement to the coronal plane. The first abducted hip was the left followed by the right one. As can be observed in Figure 4.15, the quality of the skeleton tracking output was degraded when using the kalman filter for the left hip. This was due to the filter's inability to adjust to a rapid change in the movements velocity, since the subject did not performed the abduction movement continuously in a constant velocity.

For the third sequence movement the obtained results were apparently not reliable, Figure 4.16. During the performance of the toe touch exercise the subjects were told to approximate the hands to the toe as close as possible, maintaining the knees extended. The exercise performance was evaluated by calculating the normalized hand to foot distance. This distance should be close to 1 in the beginning of the exercise, approximate to 0 as the hands touch the feet and then back to 1 as the subject returns to the initial position. The knee extension was assessed considering the knee angle, that should be close to 180º during the entire exercise. The systems incapability to return stable results for this movement was due to the severe occlusion of the hip and knee joints as the hands began to approximate the feet. Nevertheless, by comparing the raw data with the kalman filter estimate it is noteworthy the filter's ability to prevent the presence of severe outliers.

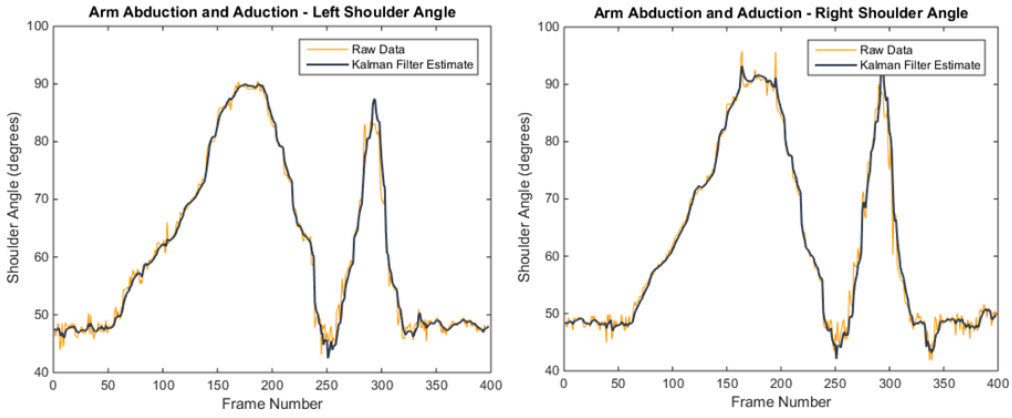

Figure 4.14: Range of motion evaluation for the shoulder angle during the abduction and adduction of the left and right arm. Results are present for both the raw data (yellow) and the Kalman filter estimate (dark blue).

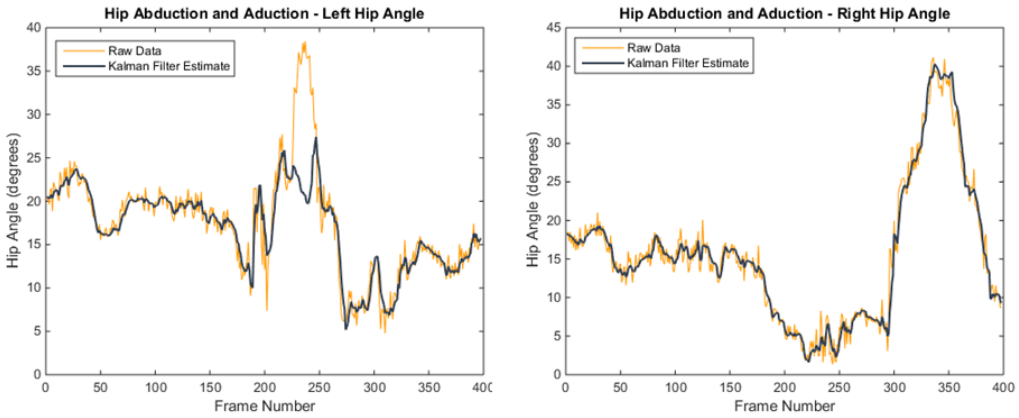

Figure 4.15: Range of motion evaluation for the hip angle during the abduction and adduction of the left and right hip. Results are present for both the raw data (yellow) and the Kalman filter estimate (dark blue).

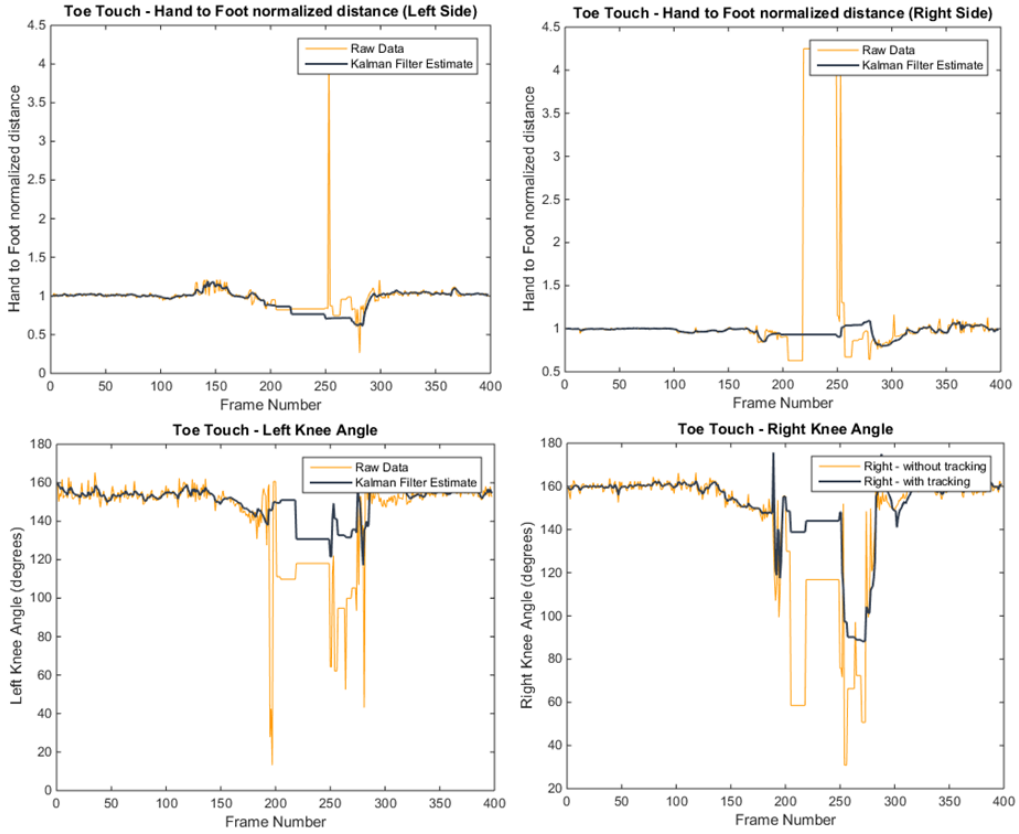

Figure 4.16: Foot to hand normalized distance (top) and knee angle (bottom) during the toe touch exercise. Results are present for both the raw data (yellow) and the Kalman filter estimate (dark blue).

In order to evaluate the performance of the proposed markerless skeleton tracking system the reported joint rotational values were compared to the ones provided by a marker based system (Qualisys AB, Gothenburg, Sweden). The joint angle trajectories of the shoulder angle during the arm abduction and adduction, both in the coronal and the saggital planes, are presented in Fig-

ures 4.17 and 4.18. For the time being, only the shoulder angle was considered for the evaluation. As shown, the trajectories of the raw data, the Kalman filter estimate and the ground-truth presented an evident correlation. This observation is based on the fact that the presented trajectories are time synchronized and follow the same pattern. Nevertheless, it can be observed that the calculation of the shoulder angle in the saggital plane (second peak) was more irregular than the one in the coronal plane (first peak). This is in accordance to previous observations that mentioned the system's difficulty in performing an accurate labelling when the movements are done outside the coronal plane. Also, when the shoulder angle was closer to 90^o, the subject's hands may occlude the shoulders and the elbows which also contributed to the observed behaviour.

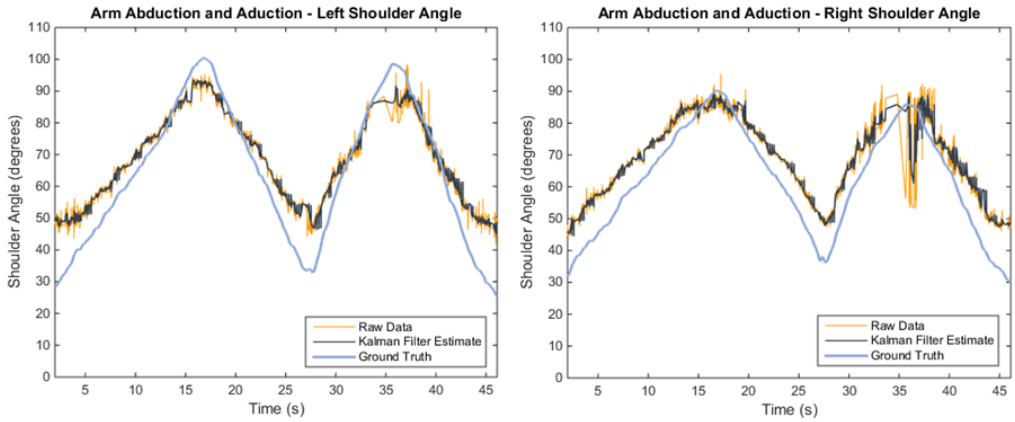

Figure 4.17: Range of motion evaluation for the shoulder angle during the abduction and adduction of the left and right arm, first in the coronal plane and then in the saggital plane, performed by the male subject. Results are present for both the raw data (yellow) and the Kalman filter estimate (dark blue). The ground-truth trajectories (light blue) were obtained using a marker based system.

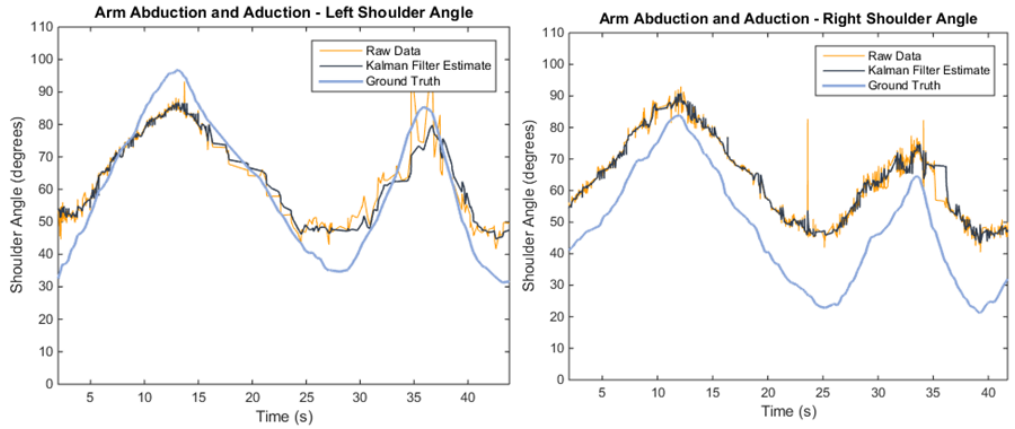

Figure 4.18: Range of motion evaluation for the shoulder angle during the abduction and adduction of the left and right arm, first in the coronal plane and then in the saggital plane, performed by the female subject. Results are present for both the raw data (yellow) and the Kalman filter estimate (dark blue). The ground-truth trajectories (light blue) were obtained using a marker based system.

The accuracy was evaluated considering the mean error (ME) [58]:

$$ME_M = \frac{1}{F}\sum_{f=1}^{F} |ST_f - O_f| \tag{4.3}$$

where M is a motion clip, F is the number of frames in a motion clip M, ST_f and O_f are the joint angles provided by the skeleton tracking system and the optical motion capture in frame f, respectively.

The mean errors of the calculation of the shoulder angle during the abduction and adduction of the left and right arm, for the three trials of both the male and the female are presented in Figure 4.19. The error was higher for the right shoulder angle of the female subject. This situation might be explained by an erroneuous marker placement, since as shown in Figure 4.18 for the right shoulder, the ground-truth trajectory presented a smaller correlation with the trajectories obtained from the developed tracking system. Although the presented trajectories of Figure 4.18 are from a single trial, the observed behaviour is consistent for the three trials. Excluding the case of the female right shoulder, the error was similar for both the male and female subjects. However, considering the small number of observations further comparisons using a larger number of subjects should be done in order to better support this observation. Considering the errors obtained with the male subject, the system's performance was within the range of other results obtained using state-of-the-art active markerless systems. In [58], a similar comparison was performed using the OpenNI, obtaining a range of errors in the calculation of the shoulder angle of 7^o to 13^o. When visually assessing the range of motion, the physical therapist evaluation normally reports an error of 10^o [58], which is higher than the overall error obtained by the proposed system (9.42^o, mean of all trials, for both the left and right side, the raw data and the Kalman filter estimate and for the male and the female subjects). Also, it is noteworthy, that the overall mean error is lower when the angle is calculated using the Kalman filter estimate instead of the raw data (9.34^o vs 9.50^o, Kalman filter estimate vs raw data, mean of all trials, for both the left and right side and for the male and the female subjects) which further sustains the Kalman filter ability to correctly estimate the motion trajectories.

Despite the higher accuracy provided by marker based system's [133], it should be noted that other factors may affect their accuracy and hence influence the comparison. These factors are related to the marker placement and soft tissue artifacts [134]. Also, it is possible that slightly discrepancies of the orientation of the stereo camera in relation to the Qualysis system may have introduced additional error between the two systems [10].

4.3 Computational Perfomance Analysis

Since in a telerehabilitation setting the real time applicability should be taken into consideration, the time performance of the proposed system was evaluated. The results are presented in Table 4.3. The first stage of the proposed pipeline (Human Body Reconstruction) was the most time consuming. In fact, these stage invalidates online processing considering the system as it is. Also,

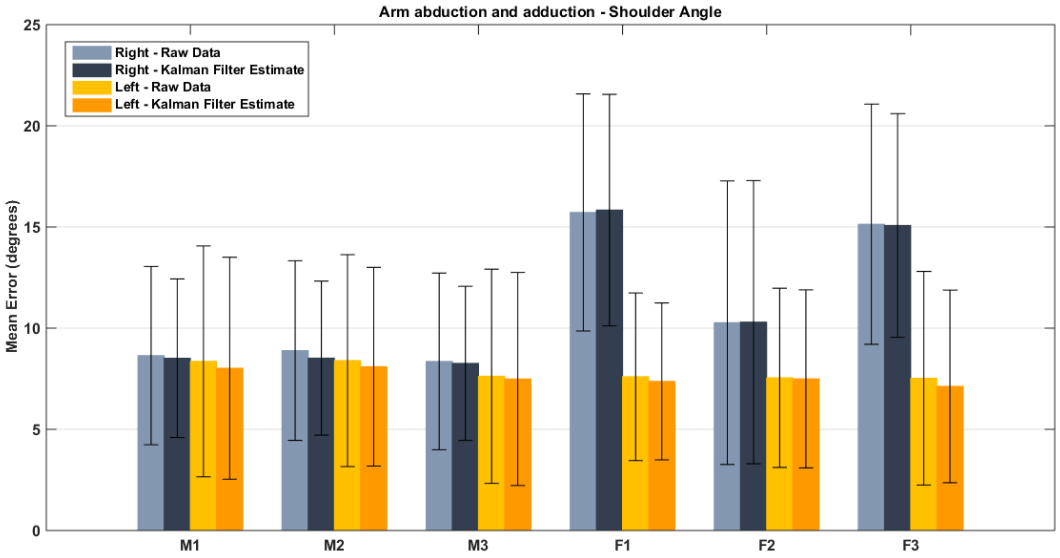

Figure 4.19: Mean error of the calculation of the shoulder angle during the abduction and adduction of the left and right arm. The presented results are the mean (and the standard deviation) for each of the three trials for both the male (M) and the female (F).

the 3D projection stage included a step of filtering. Despite visually improving the labelling outcome (Figure 4.10), the quantitative improvement achieved by the filtering stage in the quality of the obtained skeleton data was not evaluated. For this reason, it should be investigated if the improvement in the joints position accuracy compensates for the loss in time performance.

Table 4.3: Performance evaluation of the different proposed methodologies for both the Human Body Reconstruction and the Human Pose Estimation. Results are presented in seconds, as the average of 400 frames for each sequence ($\mu(\pm\sigma)$). The 3D projection stage includes the steps of filtering and background combination.

		Sequence 1	**Sequence 2**	**Sequence 3**
Human Body Reconstruction	**Disparity calculation**	0.73 ($\pm$0.76)	0.59 ($\pm$0.31)	0.69 ($\pm$0.74)
	Segmentation	3.55 ($\pm$0.85)	3.08 ($\pm$0.61)	3.22 ($\pm$0.61)
	3D Projection	17.85 ($\pm$4.71)	15.13 ($\pm$3.29)	16.53 ($\pm$2.78)
Human Pose Estimation	**Without Tracking**	0.11 ($\pm$0.02)	0.13 ($\pm$0.02)	0.12 ($\pm$0.04)
	With Tracking	0.20 ($\pm$0.02)	0.21 ($\pm$0.02)	0.17 ($\pm$0.06)

4.4 Summary

The main goal of the present study was to develop a skeleton tracking system that had as input the RGB-D information provided by a passive device, such as a stereo camera. The first step was the acquisition of the 3D human body representation. This was achieved by a process of 3D reconstruction followed by a step of segmentation and denoising. The final obtained point clouds were then used as input for the skeleton tracking system. The system used an intermediate body

part representation, from which reliable skeletons were extracted. However, the quality of the obtained skeletons was further improved with the implemented correction stage as was supported by both a qualitative and quantitative assessment. After the correction, a Kalman filter tracking approach was implemented. The returned results revealed that the filter substantially smoothed the 3D motion of the obtained skeleton data. Finally, the retrieved skeleton joints positions were used to obtain clinically relevant information that allows a quantitative assessment of the performed rehabilitation exercises. According to the obtained metrics, when the quality of the skeleton information is good, valid quantitative measures can be obtained. In order to allow the validation of the developed system, a marker based system was used to generate the needed ground-truth data. Despite revealing that the system is able to reach errors within the range of state-of-the-art markerless systems and lower than the visual evaluation done by a physical therapist, a more thorough validation study remains necessary. For that, statistical significant information should be used. Considering the time performance analysis, in the future, less time consuming methodologies should be explored to allow the achievement of real time speeds while maintaining the joints tracking accuracy.

Chapter 5

Conclusions and Future Work

5.1 Final Conclusions

The success of a rehabilitation treatment is directly related to the maintenance of a continuous activity as well as initiating the treatment as soon as possible. Nevertheless, due to the lack of resources and medical staff many patients are not receiving the appropriate treatment. This problem could be overcome with the implementation of quality telerehabilitation services. However, transferring the treatment from the hospital to the home environment demands the continuous monitoring of the patient. This monitorization is of key importance not only to ensure that the patient is performing the exercises correctly and in the exact durations and number of times prescribed, but also to guarantee a continuous feedback and guidance from the medical staff. The monitoring of the patient movements during the performance of the rehabilitation exercises could be implemented using motion-sensor technologies. Nevertheless, the used gold standard for motion tracking are marker based systems. Despite providing a high accuracy, these systems are expensive and the placement of the markers is time-consuming, needs to performed by a specialist and the acquisition can only be done in relatively large and controlled spaces which makes them unsuited for a home context.

Recently, the development of affordable and easy-to-use 3D acquisition sensors boosted their use in applications for motion tracking in a rehabilitation context. The most commonly used sensors are active whereas the advantages of passive sensors have remained unexplored.

In this thesis, the applicability of a passive device, such as a stereo camera, to analyse the human motion was explored. The first step of the proposed system was the acquisition of a 3D representation of the human body. The proposed pipeline proved to be able to reconstuct the human body with quality. This step was the most time-consuming one. Given the requirements of a real time application, the implementation of a more efficient stereo matching algorithm could potentially reduce the processing time.

The obtained 3D representation of the human body served as input for the second step that consisted in the recognition of a skeleton model. From the obtained skeleton data, quantitative measures that allow the evaluation of the performed rehabilitation exercises were extracted. The

skeleton tracking system was improved based on the kinematic relationship and anthropometrically feasible lengths between joints as well as the temporal consistency. The implemented improvements were able to considerably improve the skeleton data quality. Using as ground-truth the 3D positions of markers returned by a marker based system, the performance of the markerless system was evaluated. Results revealed that the system was able to reach errors within the range of other state-of-the-art active markerless systems and lower than the average error reported by the physical therapist. It is also noteworthy that in the context of rehabilitation an extreme accuracy is not needed. This is due to the fact that the correctness of a motion can be evaluated without being extremely precise. In fact, many of the considered exercises are evaluated based on repetitions and in comparison to a motion pattern. Also, the use of automatic systems removes the subjectivity inherent to the visual assessment done by a therapist.

Although considerable improvements are still required, the obtained results are promising. In fact, the developed methodology proved that a passive sensor, such as a stereo camera, can be used in the context of motion tracking in telerehabilitation.

Despite the fact that the focus of the present work was to explore the applicability of the use of passive devices, the improved skeleton tracking system is device agnostic. This means that it can potentially be used with any type of RGB-D information, independently of the acquisition sensor. This is not the case for most of the commonly used motion tracking open-source algorithms like the one provided by the Microsoft Kinect SDK or the OpenNI that can only be used with PrimeSense devices. The developed system could be used to explore a wider variety of acquisition sensors, such as for example, high-resolution multi-view cameras.

5.2 Future Work

Despite the potential of the obtained results, there is still room for improvement. For this reason some further developments can still be pointed.

Considering the generation of the 3D human body representation, a more suitable stereo matching algorithm could be used. Despite being able to recover the depth information properly in the presence of texture, in its absence the used algorithm performs poorly. For this reason, the development of a stereo matching algorithm less sensible to the lack of texture, such as a segmentation based one, could considerably improve the quality of the obtained disparity maps in the absence of texture. By proving a better disparity map, possibly the 3D point cloud enhancement steps could be skipped. This would considerably decrease the time needed to obtain the point clouds for the human pose estimation and tracking stage, and so improve the time performance that is a key point when developing real time applications.

Also, considering the modularity adjacent to the use of stereo cameras more views could be added. This would allow obtaining a more detailed human body and so improve the quality of the obtained skeleton data. The use of more views has the potentiality of solving some of the problems related to the ambiguity and occlusions that come from the use of only a frontal model. Also, the

acquisition of a complete 360º degrees human body model could facilitate the recovery of more exact joints positions located inside the body model.

Regarding the used intermediate body part representation from which the skeleton joints positions are obtained several improvements could potentially lead to considerably better results. The used ground truth annotated model from which the RDF classifier is learned could be refined, for example by adding a label to the shoulders or other areas of interest that are not contemplated by the current model. Also, the training data could be adapted to better suit the purpose of rehabilitation. For this, a dataset of users performing rehabilitation exercises could be used for training, which would considerably improve the recognition task. Many of the rehabilitation exercises used in the clinical practice are aided by the presence of common objects such as chairs, balls and elastics. However, the presence of this objects hampers the task of body recognition. The inclusion of some of this objects in the training dataset could possibly overcome this limitation. Furthermore, instead of using a single slim male model to create the dataset, a wider variety of subjects should be used, varying for example the gender, age, height, weight, hair and clothing, including as well subjects with amputations and physical handicaps. Also, rather than generating the synthetic depth information considering only a straight-on, chest-height camera angle, a larger variety of camera positions and orientations could be considered. As well, the synthetic obtained depth maps could be improved by adding sensor-dependent noise models.

The proposed tracking methodology was implemented considering each joint individually. Given the articulated nature of the human skeleton, the tracking outcome would substantially benefit from incorporating the kinematic relationship between each joint in the used measurement model. Likewise, the Kalman filter is suited to model linear noise. Given the unpredictability of the human motion a more robust non linear filter, such as the Extended Kalman Filter or the Unscented Kalman filter could improve the tracking outcome.

Despite the preliminary evaluation presented using the information provided by a marker based system, the gathered data should be used to extract a wider range of quantitative measures. This will allow a more consistent evaluation of the proposed system.

Finally, the system's performance should be tested in a wider range of the population, considering not only healthy as unhealthy patients, exploring functional and clinically relevant movements used in the context of rehabilitation. Also, it would be interesting to assess the system's performance in evaluating the progression and treatment of movement debilitating specific diseases such as Parkinson.

Appendix A

Acquisition Protocol

A.1 Acquisition

A.1.1 Requirements

A computer must be connected to the Bumblebee2.

System requirements:

- Windows or Linux (32-bit or 64-bit)
- 3.1 GHz or equivalent CPU
- 2 GB RAM or more
- 200 MB hard drive space
- FireWire IEEE-1394a port
- FlyCapture SDK and Triclops SDK
- Microsoft Visual Studio 2010 or 2008
- 4.5 meter, 6-pin to 6-pin, IEEE-1394 cable

A.1.2 Camera Position

The camera should be positioned as close as possible from the subject ensuring that the entire subject is observed within the camera range. Considering a subject with around 1.70 m the distance between the subject and the stereo camera should be 2.7-3.5 m. The distance from the floor should be around 0.9 m. The use of a tripod is advisable.

Only the subject should be in the detection's range of the sensor.

A.1.3 Acquisition Parameters

Raw images are acquired with a 640x480 resolution in Raw16 (16 bpp) pixel interleaved format (the first byte is from the left camera and the second from the right). A maximum frame rate of about 48fps can be obtained. In order to achieve better acquisition times only raw images are acquired (unpacking and rectification is performed later during the processing stage).

A.1.4 Room Environment

The room were the acquisition is performed should have the adequate dimensions to allow a full body tracking. A moderate light should be used allowing that the subject body is evenly lit. Side and back lighting should be avoided. Reflective backgrounds should also be avoided.

A.1.5 Subject

If possible, the subject should be young, healthy, injury free and without history of neurological conditions or medication that may influence their postural control.

The subject should wear textured, colourful and tight fitting clothes. Dark and plain clothes should be avoided. In case dark clothes are used, a synthetic texture should be created.

The acquisition is performed with the subject (initially) in a standing position.

During the acquisition the patient should perform the set of movements described in section A.1.6.

A.1.6 Movements

All movements, described in Table A.1 and presented in Figure A.1, should be performed with identical instructions:

- Start from a standing position;
- Slowly perform the required exercise;
- Maintain the goal position for approximately 5 seconds;
- Slowly return to the initial position.

Table A.1: Detailed description of the rehabilitation exercises performed by the subjects during image acquisition.

Exercise	Description
1	Arm abduction and adduction in the coronal plane followed by arm abduction and adduction in the sagittal plane.
2	Hip abduction and adduction in the coronal plane with the knee extended (left leg followed by the right leg).
3	Toe touch: Movement of the hands from the sides of the trunk in the direction of the toes.

A.2 Data

Using FlyCapture SDK and Triclops SDK raw images are acquired. If necessary, de-interleaved and unprocessed images for the right and left view can be simultaneously acquired. However, that slows the acquisition process (from around 20 fps to 1.5 fps).

Very important: prior to each acquisition a calibration (.cal) file containing the camera and acquisition parameters should be saved.

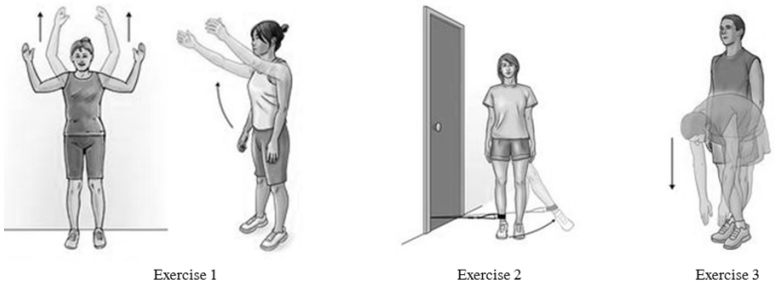

Figure A.1: Rehabilitation exercises performed by the subjects during image acquisition. (Adapted from [101].)

A.2.1 Files Organization

All the gathered information regarding a sequence movement is stored in a folder with the name of the respective sequence.

Raw images are stored as: "`raw_sequencename_f_timestamp.raw`".

If acquired, de-interleaved and unprocessed images are store as:

"`raw_view_sequencename_f_timestamp.extension`" where view=left,right, extension=pgm,ppm if grayscale or rgb images are acquired, respectively.

sequencename is the name of the capture sequence movement, timestamp is the time in microseconds since epoch in which the frame is acquired.

Appendix B

Joints Position Tracking Evaluation

In this appendix, additional results regarding the evaluation of the joints position tracking using a Kalman filter, discussed in Chapter 4, are presented.

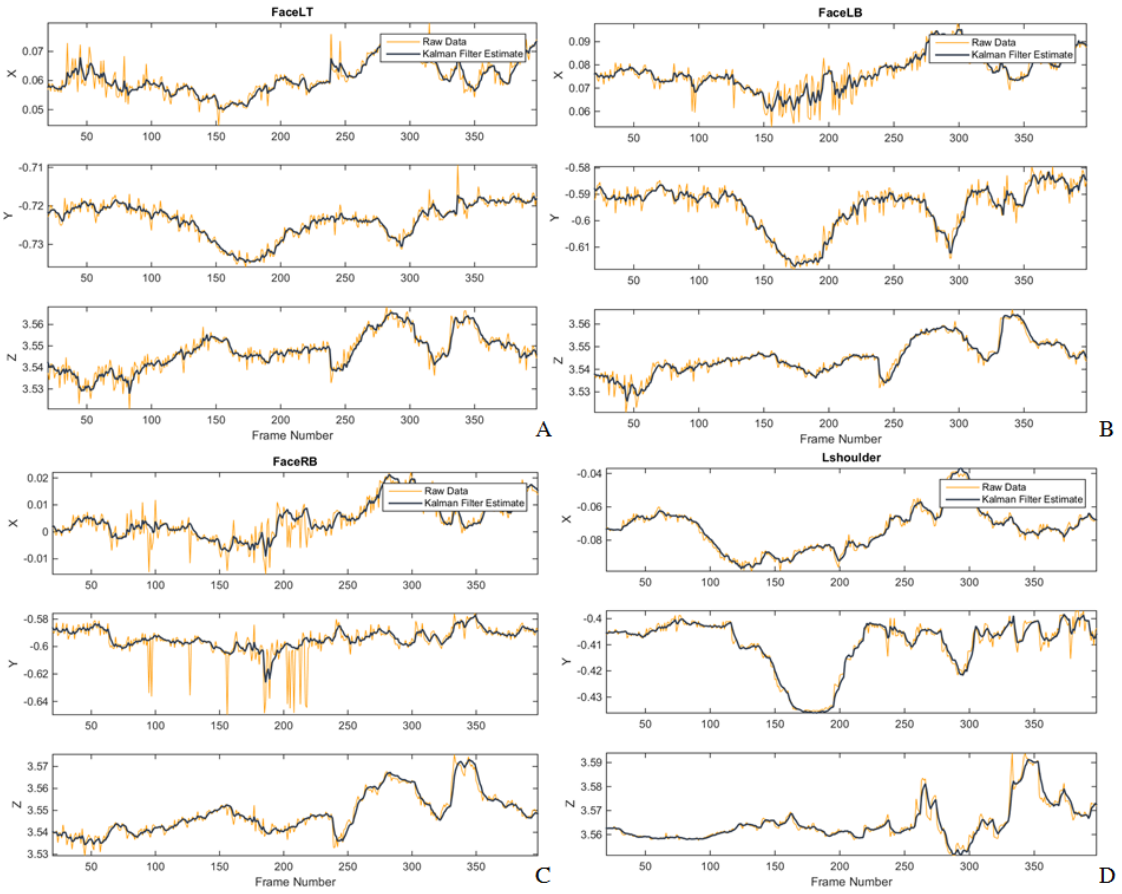

Figure B.1: Comparison of the Kalman filter estimate (dark blue) and the raw data (yellow), in meters, for the x (top), y (middle) and z (bottom) trajectories. (A) Face left top, (B) face left bottom, (C) face right bottom and (D) left shoulder. When the skeleton tracking system is not able to return a joint position, the x, y and z coordinates are marked as -1.

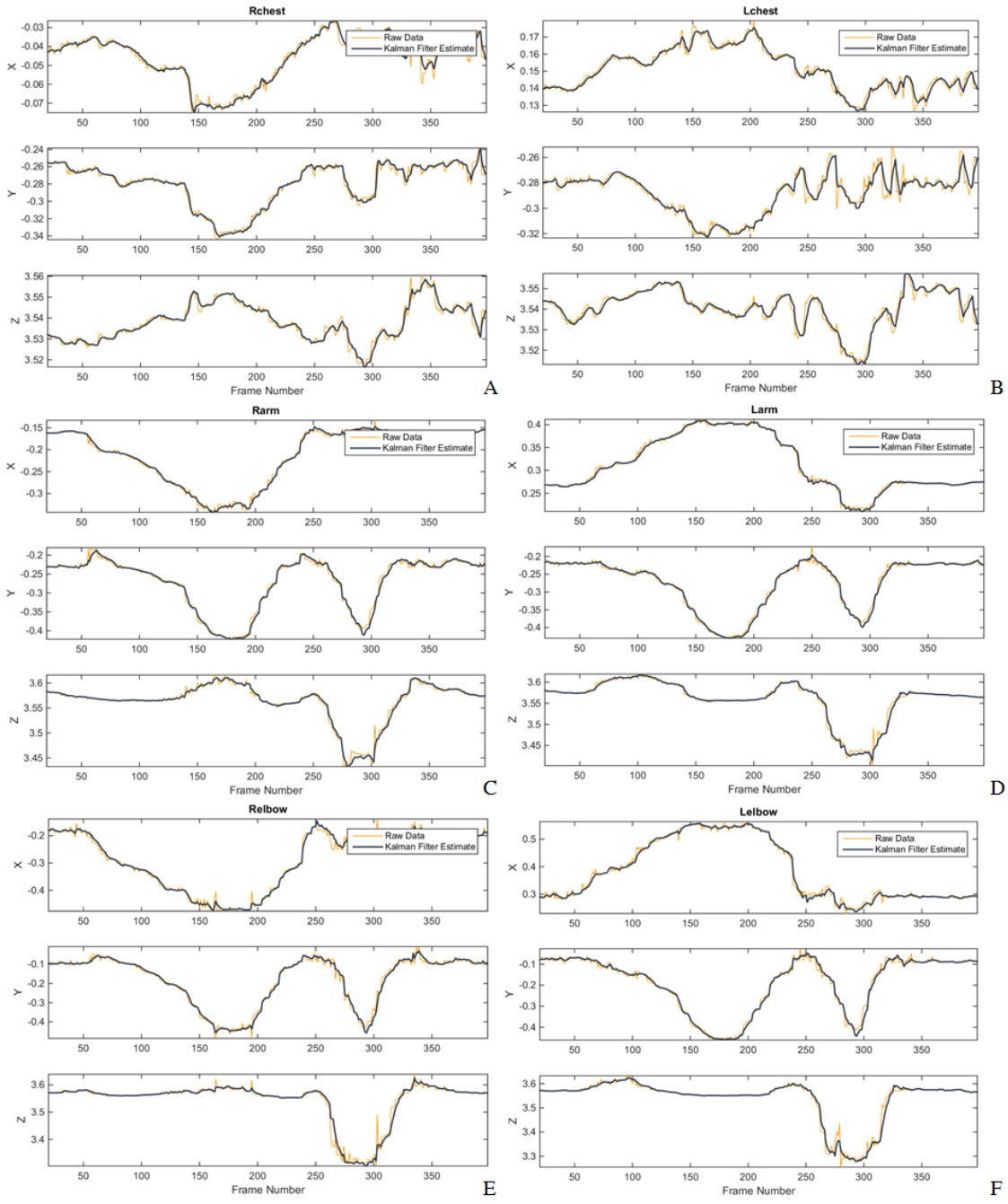

Figure B.2: Comparison of the Kalman filter estimate (dark blue) and the raw data (yellow), in meters, for the x (top), y (middle) and z (bottom) trajectories. (A) Right chest, (B) left chest, (C) right arm, (D) left arm, (E) right elbow and (F) left elbow. When the skeleton tracking system is not able to return a joint position, the x, y and z coordinates are marked as -1.

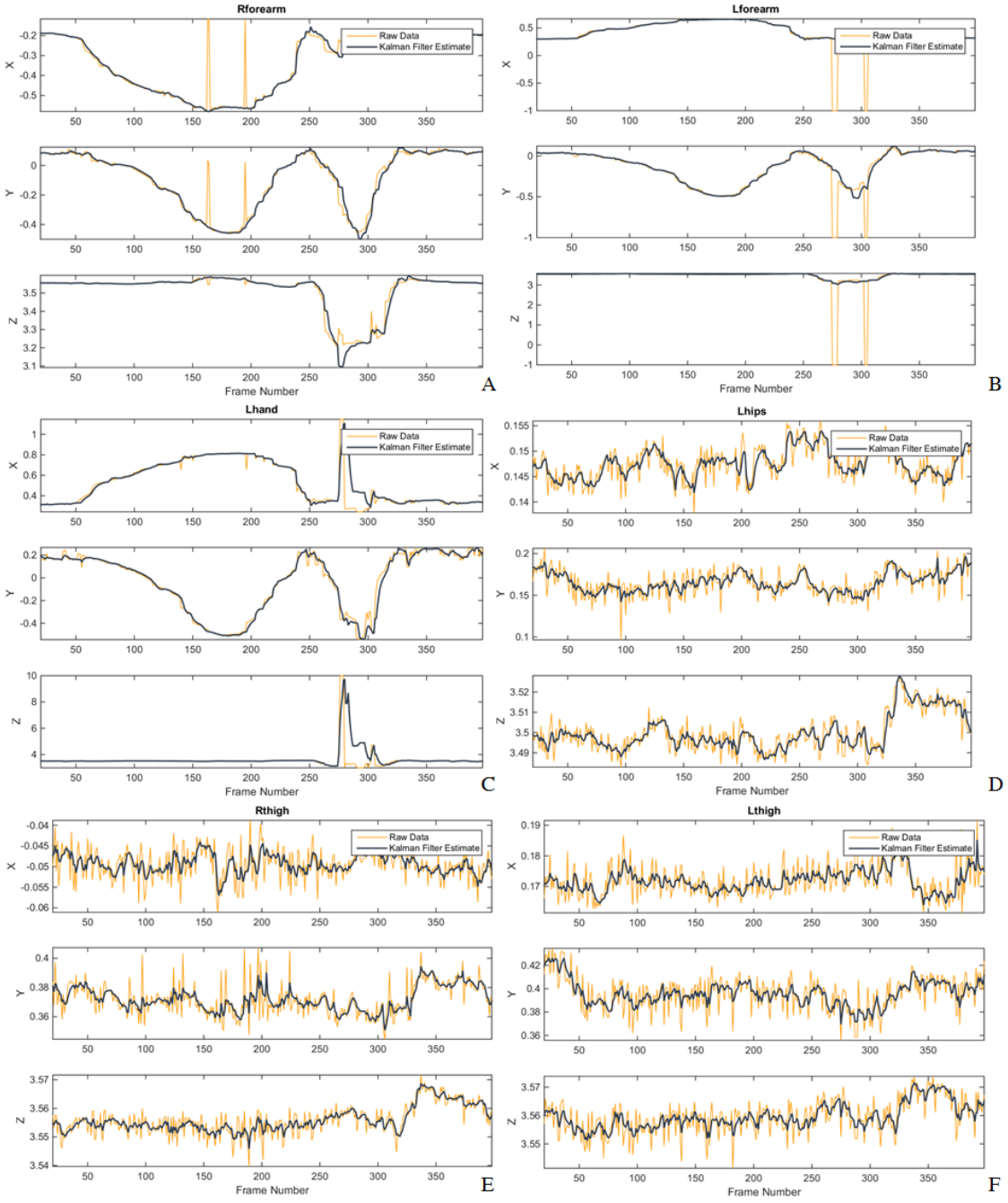

Figure B.3: Comparison of the Kalman filter estimate (dark blue) and the raw data (yellow), in meters, for the x (top), y (middle) and z (bottom) trajectories. (A) Right forearm, (B) left forearm, (C) left hand, (D) left hip, (E) right thigh and (F) left thigh. When the skeleton tracking system is not able to return a joint position, the x, y and z coordinates are marked as -1.

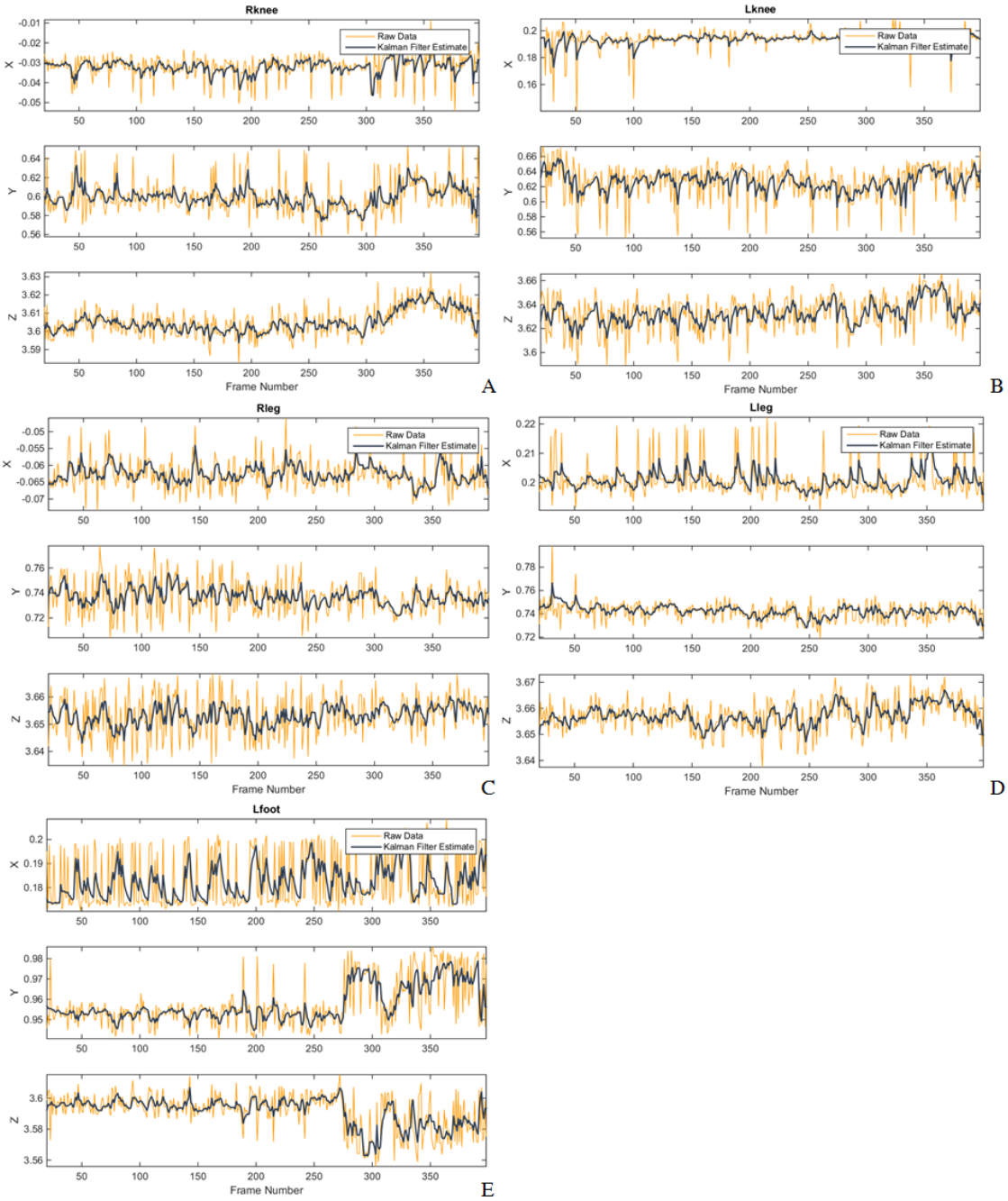

Figure B.4: Comparison of the Kalman filter estimate (dark blue) and the raw data (yellow), in meters, for the x (top), y (middle) and z (bottom) trajectories. (A) Right knee, (B) left knee, (C) right leg, (D) left leg and (E) left foot. When the skeleton tracking system is not able to return a joint position, the x, y and z coordinates are marked as -1.

Appendix C

Point Cloud Segmentation for Validation

Given the differences in the acquisition environment of the stereo images used for the validation, the proposed pipeline for the segmentation (Figure 3.10) was unable to properly segment the subject. For that, a new segmentation pipeline was envisioned and is described in Figure C.1.

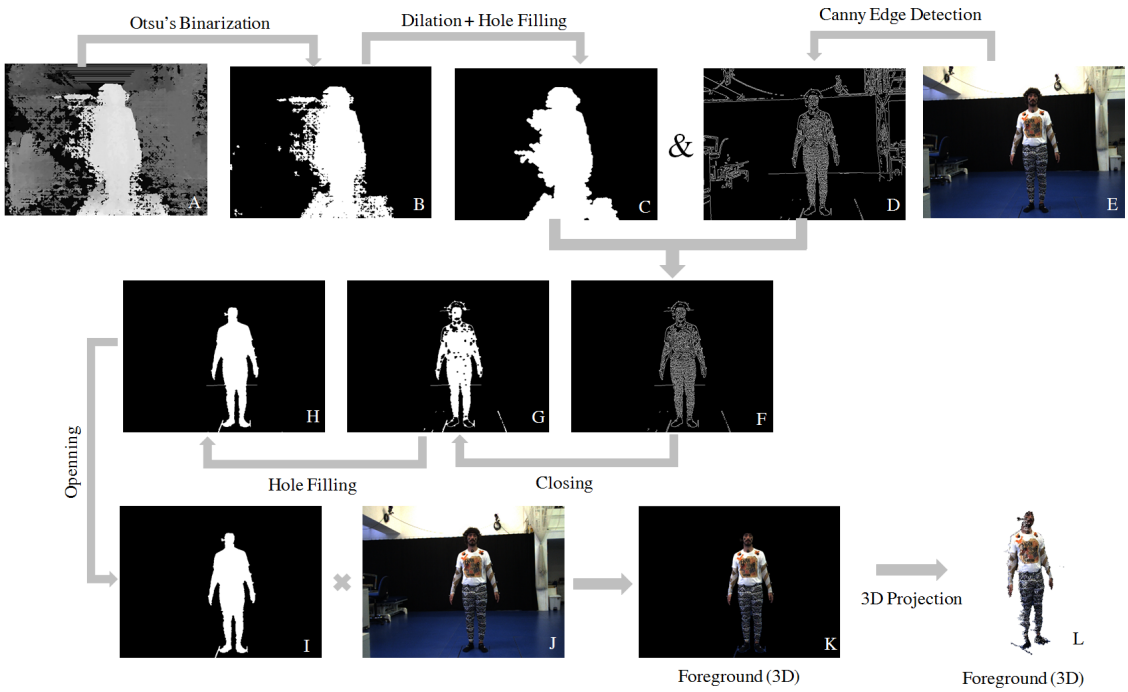

Figure C.1: Foreground segmentation pipeline. The disparity image (A) was used to obtain a rough estimate of the subject's position (B) through Otsu's binarization, that was further improved by using a dilation and hole filling strategy (C). The obtained mask was then combined with the binary image (D), acquired by using the Canny edge detector applied onto the initial RGB image (E). The resulting mask (E) was then enhanced based on the use of morphological operations and hole filling methodologies (G-H). The final mask (I) was combined with the original RGB image (J) and used to obtain the refined RGB segmented subject (K) that was projected to 3D (L).

When comparing to the previous described segmentation pipeline, the main difference relies on the way the binary mask is enhanced. In the initial pipeline, the subjects silhouette was enhanced based on the GrabCut method. However, given the colour resemblance between the background and foreground (mainly noticeable on the lower members) the GrabCut method failed to be efficient. For that reason the subject's silhouette was retrieved by using the the Canny edge detector [135]. After the edge detection the obtained mask was further enhanced following a closing morphological operation and a hole filling strategy. Such as in the initial pipeline, the final mask was used to obtain the refined RGB segmented subject that was projected to 3D.

References

[1] D. Kairy, P. Lehoux, C. Vincent, and M. Visintin. A systematic review of clinical outcomes, clinical process, healthcare utilization and costs associated with telerehabilitation. *Disability and Rehabilitation*, 31(6):427–447, 2009.

[2] World report on disability. Report, World Health Organization, 2011.

[3] M. Rogante, M. Grigioni, D. Cordella, and C. Giacomozzi. Ten years of telerehabilitation: A literature overview of technologies and clinical applications. *NeuroRehabilitation*, 27(4):287–304, 2010.

[4] B. Parmanto and A. Saptono. Telerehabilitation: State-of-the-art from an informatics perspective. *International Journal of Telerehabilitation*, 1(1):73–84, 2009.

[5] M. Zampolini, E. Todeschini, M. B. Guitart, H. Hermens, S. Ilsbroukx, V. Macellar, R. Magni, M. Rogante, S. S. Marches, M. Vollenbroek, and C. Giacomozzi. Telerehabilitation: present and future. *Ann. Ist. Super. Sanita*, 44(2):125–134, 2008.

[6] H. Zhou and H. Hu. Human motion tracking for rehabilitation-a survey. *Biomedical Signal Processing and Control*, 3:1–18, 2008.

[7] M. R. Schmeler, R. M. Schein, M. McCue, and K. Betz. Telerehabilitation clinical and vocational applications for assistive technology: Research, opportunities, and challenges. *International Journal of Telerehabilitation*, 1(1):59–72, 2009.

[8] W. Zhao, D.D. Espy, M.A. Reinthal, and Hai Feng. A feasibility study of using a single kinect sensor for rehabilitation exercises monitoring: A rule based approach. In *Computational Intelligence in Healthcare and e-health (CICARE), 2014 IEEE Symposium on*, pages 1–8, Dec 2014.

[9] B. Bonnechère, B. Jansen, P. Salvia, H. Bouzahouene, L. Omelina, F. Moiseev, V. Sholukha, J. Cornelis, M. Rooze, and S. Van Sint Jan. Validity and reliability of the kinect within functional assessment activities: Comparison with standard stereophotogrammetry. *Gait & Posture*, 39(1):593 – 598, 2014.

[10] B. Galna, G. Barry, D. Jackson, D. Mhiripiri, P. Olivier, and L. Rochester. Accuracy of the microsoft kinect sensor for measuring movement in people with parkinson's disease. *Gait & Posture*, 39(4):1062 – 1068, 2014.

[11] L. Chen, H. Wei, and J. Ferryman. A survey of human motion analysis using depth imagery. *Pattern Recognition Letters*, 34(15):1995 – 2006, 2013. Smart Approaches for Human Action Recognition.

[12] A. Scano, M. Caimmi, M. Malosio, and L. M. Tosatti. Using kinect for upper-limb functional evaluation in home rehabilitation: A comparison with a 3d stereoscopic passive marker system. In *Biomedical Robotics and Biomechatronics (2014 5th IEEE RAS EMBS International Conference on*, pages 561–566, Aug 2014.

[13] R. A. Clark, Y. Pua, K. Fortin, C. Ritchie, K. E. Webster, L. Denehy, and A. L. Bryant. Validity of the microsoft kinect for assessment of postural control. *Gait & Posture*, 36(3):372 – 377, 2012.

[14] D. González-Ortega, F.J. Díaz-Pernas, M. Martínez-Zarzuela, and M. Antón-Rodríguez. A kinect-based system for cognitive rehabilitation exercises monitoring. *Computer Methods and Programs in Biomedicine*, 113(2):620 – 631, 2014.

[15] N. Lazaros, G. C. Sirakoulis, and A. Gasteratos. Review of stereo vision algorithms: From software to hardware. *International Journal of Optomechatronics*, 2(4):435–462, 2008.

[16] K. Buys, J. Hauquier, C. Cagniart, T. Tuytelaars, and J. De Schutter. Virtual data generation based on a human model for machine learning applications. In *Proceedings of the international Digital Human Modeling Symposium*, pages 1–9, 2013.

[17] F. Alhwarin, A. Ferrein, and I. Scholl. Ir stereo kinect: Improving depth images by combining structured light with ir stereo. In *PRICAI 2014: Trends in Artificial Intelligence*, volume 8862 of *Lecture Notes in Computer Science*, pages 409–421. Springer International Publishing, 2014.

[18] J. M. Winters. Telerehabilitation research: Emerging opportunities. *Annu. Rev. Biomed. Eng.*, 4:287–320, 2002.

[19] D. M Brennan and L. M. Barker. Human factors in the development and implementation of telerehabilitation systems. *Journal of Telemedicine and Telecare*, 14(2):55–58, 2008.

[20] M. McCue, A. Fairman, and M. Pramuka. Enhancing quality of life through telerehabilitation. *Phys Med Rehabil Clin N Am*, 21(1):195–205, 2010.

[21] A. G. Ekeland, A. Bowes, and S. Flottorp. Effectiveness of telemedicine: A systematic review of reviews. *International Journal of Medical Informatics*, 79(11):736 – 771, 2010.

[22] M. Tousignant, P. Boissy, H. Corriveau, and H. Moffet. In home telerehabilitation for older adults after discharge from an acute hospital or rehabilitation unit: A proof-of-concept study and costs estimation. *Disabil Rehabil: Assist Technol*, 1(4):209–216, 2006.

[23] H. Kortke, H. Stromeyer, A. Zittermann, N. Buhr, E. Zimmermann, E. Wienecke, and R. Korfer. New east-westfalian postoperative therapy concept: a telemedicine guide for the study of ambulatory rehabilitation of patients after cardiac surgery. *Telemed J E Health*, 12(4):475–483, 2006.

[24] S. V. Rojas and M. P. Gagnon. A systematic review of the key indicators for assessing telehomecare cost-effectiveness. *Telemed J E Health*, 14(9):896–904, 2008.

[25] K. Nakamura, T. Takano, and C. Akao. The effectiveness of videophones in home healthcare for the elderly. *Medical Care*, 37(2):117–125, 1999.

[26] M Pramuka and L. van Roosmalen. Telerehabilitation technologies: Accessibility and usability. *Int J Telerehabil.*, 1(1):85–98, 2008.

[27] J.P.S. Cunha, B. Cunha, A.S. Pereira, W. Xavier, N. Ferreira, and L. Meireles. Vital-jacket: A wearable wireless vital signs monitor for patients' mobility in cardiology and sports. 1(1):1–2, March 2010.

[28] M. K. Holden. Virtual environments for motor rehabilitation: Review. *Cyber Psychology & Behavior*, 8(3):187–211, 2005.

[29] T. C.S. Azevedo, J. M. R.S. Tavares, and M. A.P. Vaz. Three-dimensional reconstruction and characterization of human external shapes from two-dimensional images using volumetric methods. *Computer Methods in Biomechanics and Biomedical Engineering*, 13(3):359–369, 2010.

[30] W. Yu and B. Xu. A portable stereo vision system for whole body surface imaging. *Image Vis Comput*, 28(4):605–13, 2010.

[31] P. Costa, H. Zolfagharnasab, J. P. Monteiro, J. S. Cardoso, and H. P. Oliveira. 3d reconstruction of body parts using rgb-d sensors: Challenges from a biomedical perspective. In *Proceedings of the 5th International Conference on 3D Body Scanning Technologies*, pages 378–389, 2014.

[32] J. C. K. Wells, A. Ruto, and P. Treleaven. Whole-body three-dimensional photonic scanning: a new technique for obesity research and clinical practice. *Int J Obes*, 32(2):232–238, 2007.

[33] T. J.J. Maal, B. van Loon, J. M. Plooij, F. Rangel, A. M. Ettema, W. A. Borstlap, and S. J. Bergé. Registration of 3-dimensional facial photographs for clinical use. *Journal of Oral and Maxillofacial Surgery*, 68(10):2391 – 2401, 2010.

[34] L. van Gool T. Moons and M. Vergauwen. 3d reconstruction from multiple images part 1: Principles. *Foundations and Trends® in Computer Graphics and Vision*, 4(4):287–404, 2008.

[35] S. X.M. Yang, M. S. Christiansen, P. K. Larsen, T. Alkjær, T. B. Moeslund, E. B. Simonsen, and N. Lynnerup. Markerless motion capture systems for tracking of persons in forensic biomechanics: an overview. *Computer Methods in Biomechanics and Biomedical Engineering: Imaging and Visualization*, 2(1):46–65, 2014.

[36] H. Yang and S. Lee. Reconstruction of 3d human body pose from stereo image sequences based on top-down learning. *Pattern Recognition*, 40(11):3120 – 3131, 2007.

[37] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *Mixed and Augmented Reality (ISMAR), 2011 10th IEEE International Symposium on*, pages 127–136, 2011.

[38] S. Schuon, C. Theobalt, J. Davis, and S. Thrun. Lidarboost: Depth superresolution for tof 3d shape scanning. *In Proc. of IEEE CVPR 2009*, 2009.

[39] J. Choi, Y. Choe, and Y. Kim. Sparsity based depth estimation and hole-filling algorithm for 2d to 3d video conversion. In *Signals and Electronic Systems (ICSES), 2012 International Conference*, pages 1–4, Sept 2012.

[40] L. Po, S. Zhang, X. Xu, and Y. Zhu. A new multidirectional extrapolation hole-filling method for depth-image-based rendering. In *Image Processing (ICIP), 2011 18th IEEE International Conference on*, pages 2589–2592, Sept 2011.

[41] J. Gautier, O. Le Meur, and C. Guillemot. Depth-based image completion for view synthesis. In *3DTV Conference: The True Vision - Capture, Transmission and Display of 3D Video (3DTV-CON), 2011*, pages 1–4, May 2011.

[42] J. Ziegler, K. Nickel, and R. Stiefelhagen. Tracking of the articulated upper body on multi-view stereo image sequences. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 774–781, June 2006.

[43] Y. Cui, W. Changz, T. Nolly, and D. Strickery. Kinectavatar: Fully automatic body capture using a single kinect. pages 133–147, 2012.

[44] J. Tong, J. Zhou, L. Liu, Z. Pan, and H. Yan. Scanning 3d full human bodies using kinects. *Visualization and Computer Graphics, IEEE Transactions on*, 18(4):643–650, 2012.

[45] J. Tong, M. Zhang, X. Xiang, H. Shen, H. Yan, and Z. Chen. 3d body scanning with hairstyle using one time-of-flight camera. *Comput. Animat. Virtual Worlds*, 22(2-3):203–211, 2011.

[46] M. Böhme, M. Haker, T. Martinetz, and E. Barth. Shading constraint improves accuracy of time-of-flight measurements. *Computer Vision and Image Understanding*, 114(12):1329–1335, 2010.

[47] Y. Cui, S. Schuon, D. Chan, S. Thrun, and C. Theobalt. 3d shape scanning with a time-of-flight camera. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 1173–1180, June 2010.

[48] J. Cho, K. Sung-Yeol, Y. Ho, and K. H. Lee. Dynamic 3d human actor generation method using a time-of-flight depth camera. *Consumer Electronics, IEEE Transactions on*, 54(4):1514–1521, 2008.

[49] H. A. M. Daanen and F. B. Ter Haar. 3d whole body scanners revisited. *Displays*, 34(4):270 – 275, 2013.

[50] K. Miyazawa and T. Aoki. A robot-based 3d body scanning system using passive stereo vision. In *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on*, pages 305–308, 2008.

[51] P. J. Besl and N. D. McKay. A method for registration of 3-d shapes. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 14(2):239–256, Feb 1992.

[52] G. Somanath, S. Cohen, B. Price, and C. Kambhamettu. Stereo+kinect for high resolution stereo correspondences. In *3D Vision - 3DV 2013, 2013 International Conference on*, pages 9–16, June 2013.

[53] R. Nair, K. Ruhl, F. Lenzen, S. Meister, H. Schäfer, C. Garbe, M. Eisemann, M. Magnor, and D. Kondermann. A survey on time-of-flight stereo fusion. In *Time-of-Flight and Depth Imaging. Sensors, Algorithms, and Applications*, volume 8200 of *Lecture Notes in Computer Science*, pages 105–127. 2013.

[54] J. Zhu, L. Wang, R. Yang, J. E. Davis, and Z. Pan. Reliability fusion of time-of-flight depth and stereo geometry for high quality depth maps. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 33(7):1400–1414, 2011.

[55] R. Kindermann and J. L. Snell. *Markov Random Fields and Their Applications*. AMS, 1980.

[56] W. Jia, W. Yi, J. Saniie, and E. Oruklu. 3d image reconstruction and human body tracking using stereo vision and kinect technology. In *Electro/Information Technology (EIT), 2012 IEEE International Conference on*, pages 1–4, 2012.

[57] Š. Obdržálek, G. Kurillo, J. Han, T. Abresch, and R. Bajcsy. Real-time human pose detection and tracking for tele-rehabilitation in virtual reality. *Stud Health Technol Inform*, 173:320–324, 2012.

[58] A. Fernandez-Baena, A. Susin, and X. Lligadas. Biomechanical validation of upper-body and lower-body joint movements of kinect motion capture data for rehabilitation treatments. In *Intelligent Networking and Collaborative Systems (INCoS), 2012 4th International Conference on*, pages 656–661, Sept 2012.

[59] T. B. Moeslund, A. Hilton, and V. Krüger. A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding*, 104(2–3):90 – 126, 2006. Special Issue on Modeling People: Vision-based understanding of a person's shape, appearance, movement and behaviour.

[60] P. Kohli, J. Rihan, M. Bray, and P. H. S. Torr. Simultaneous segmentation and pose estimation of humans using dynamic graph cuts. *International Journal of Computer Vision*, 79(3):285–298, 2008.

[61] K. Ogawara, X. Li, and K. Ikeuchi. Marker-less human motion estimation using articulated deformable model. In *Robotics and Automation, 2007 IEEE International Conference on*, pages 46–51, April 2007.

[62] D. Anguelov, P. Srinivasan, D. Koller, S. Thrun, J. Rodgers, and J. Davis. Scape: Shape completion and animation of people. *ACM Trans. Graph.*, 24(3):408–416, July 2005.

[63] K. Pearson. On lines and planes of closest fit to systems of points in space. *Philosophical Magazine Series 6*, 2(11):559–572, 1901.

[64] P. Guan, A. Weiss, A.O. Balan, and M.J. Black. Estimating human shape and pose from a single image. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1381–1388, Sept 2009.

[65] S. Zhou, H. Fu, L. Liu, D. Cohen, and X. Han. Parametric reshaping of human bodies in images. *ACM TOG (Proc. SIGGRAPH)*, 29(4):126–136, 2010.

[66] A. Jain, T. Thormählen, H. Seidel, and C. Theobalt. Moviereshape: Tracking and reshaping of humans in videos. *ACM Trans. Graph.*, 2010.

[67] D. A. Hirshberg, M. Loper, E. Rachlin, and M. J. Black. Coregistration: Simultaneous alignment and modeling of articulated 3d shape. In *European Conference on Computer Vision*, pages 242–255, 2012.

[68] A. Weiss, D. Hirshberg, and M.J. Black. Home 3d body scans from noisy image and range data. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1951–1958, Nov 2011.

[69] M. Ye, X. Wang, R. Yang, L. Ren, and M. Pollefeys. Accurate 3d pose estimation from a single depth image. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 731–738, Nov 2011.

[70] S. Corazza, E. Gambaretto, L. Mündermann, and T. P. Andriacchi. Automatic generation of a subject-specific model for accurate markerless motion capture and biomechanical applications. *IEEE Transactions on Biomedical Engineering*, 57:806–812, 2010.

[71] S. Corazza, L. Mündermann, E. Gambaretto, G. Ferrigno, and T. P. Andriacchi. Markerless motion capture through visual hull, articulated icp and subject specific model generation. *International Journal of Computer Vision*, 87(1-2):156–169, 2009.

[72] D. Grest, J. Woetzel, and R. Koch. Nonlinear body pose estimation from depth images. In *Pattern Recognition*, volume 3663 of *Lecture Notes in Computer Science*, pages 285–292. 2005.

[73] J. Chen, X. Wu, M. Y. Wang, and F. Deng. Human body shape and motion tracking by hierarchical weighted icp. In *Advances in Visual Computing*, volume 6939 of *Lecture Notes in Computer Science*, pages 408–417. 2011.

[74] A. Myronenko and X. Song. Point set registration: Coherent point drift. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(12):2262–2275, Dec 2010.

[75] L. A. Schwarz, A. Mkhitaryan, D. Mateus, and N. Navab. Human skeleton tracking from depth data using geodesic distances and optical flow. *Image and Vision Computing*, 30(3):217 – 226, 2012. Best of Automatic Face and Gesture Recognition 2011.

[76] Z. Li and D. Kulic. A stereo camera based full body human motion capture system using a partitioned particle filter. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 3428–3434, Oct 2010.

[77] E. Yeguas-Bolivar, R. Muñoz-Salinas, R. Medina-Carnicer, and A. Carmona-Poyato. Comparing evolutionary algorithms and particle filters for markerless human motion capture. *Applied Soft Computing*, 17(0):153 – 166, 2014.

[78] N. Hansen. The cma evolution strategy: A comparing review. In *Towards a New Evolutionary Computation*, volume 192 of *Studies in Fuzziness and Soft Computing*, pages 75–102. 2006.

[79] K. Price, R. M. Storn, and J. A. Lampinen. *Differential Evolution: A Practical Approach to Global Optimization (Natural Computing Series)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2005.

[80] J. Kennedy and R. Eberhart. Particle swarm optimization. In *Neural Networks, 1995. Proceedings., IEEE International Conference on*, volume 4, pages 1942–1948, Nov 1995.

[81] J. Deutscher and I. Reid. Articulated body motion capture by stochastic search. *International Journal of Computer Vision*, 61(2):185–205, 2005.

[82] B. Jan, F. Engstler, and M. Beetz. Evaluation of hierarchical sampling strategies in 3d human pose estimation. In *Proceedings of the 19th British Machine VisionConference, BMVC'08*, pages 1–10, 2008.

[83] D. Michel, C. Panagiotakis, and A. A. Argyros. Tracking the articulated motion of the human body with two rgbd cameras. *Machine Vision and Applications*, 26(1):41–54, 2015.

[84] J. Shotton, A. Fitzgibbon, M. Cook, T. Sharp, M. Finocchio, R. Moore, A. Kipman, and A. Blake. Real-time human pose recognition in parts from a single depth image. In *CVPR*. IEEE, June 2011.

[85] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[86] J. Han, L. Shao, D. Xu, and J. Shotton. Enhanced computer vision with microsoft kinect sensor: A review. *IEEE Trans. Cybernetics*, 43(5), October 2013.

[87] D. Dinh, M. Lim, N. Thang, S. Lee, and T. Kim. Real-time 3d human pose recovery from a single depth image using principal direction analysis. *Applied Intelligence*, 41(2):473–486, 2014.

[88] M. Jiu, C. Wolf, G. Taylor, and A. Baskurt. Human body part estimation from depth images via spatially-constrained deep learning. *Pattern Recognition Letters*, 50(0):122 – 129, 2014.

[89] B. Holt, E. Ong, and R. Bowden. Accurate static pose estimation combining direct regression and geodesic extrema. In *Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on*, pages 1–7, April 2013.

[90] R. Girshick, J. Shotton, P. Kohli, A. Criminisi, and A. Fitzgibbon. Efficient regression of general-activity human poses from depth images. In *ICCV*. IEEE, October 2011.

[91] J. Taylor, J. Shotton, T. Sharp, and A. Fitzgibbon. The vitruvian manifold: Inferring dense correspondences for one-shot human pose estimation. In *Proc. CVPR*. IEEE, June 2012.

[92] W. Shen, K. Deng, X. Bai, T. Leyvand, B. Guo, and Z. Tu. Exemplar-based human action pose correction and tagging. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1784–1791, June 2012.

[93] L. Zhou, Z. Liu, H. Leung, and H. P. H. Shum. Posture reconstruction using kinect with a probabilistic model. In *Proceedings of the 20th ACM Symposium on Virtual Reality Software and Technology*, VRST '14, pages 117–125. ACM, 2014.

[94] J. Quinonero-candela, C. E. Rasmussen, and R. Herbrich. A unifying view of sparse approximate gaussian process regression. *Journal of Machine Learning Research*, 6:1939–1959, 2005.

[95] I. Culjak, D. Abram, T. Pribanic, H. Dzapo, and M. Cifrek. A brief introduction to opencv. In *MIPRO, 2012 Proceedings of the 35th International Convention*, pages 1725–1730, May 2012.

[96] G. R. Bradski and A. Kaehler. *Learning Opencv, 1st Edition*. O'Reilly Media, Inc., 1 edition, 2008.

[97] R.B. Rusu and S. Cousins. 3d is here: Point cloud library (pcl). pages 1–4, May 2011.

[98] About - what is pcl? Accessed:26-08-2015. URL: `http://pointclouds.org/about/`.

[99] A. Shingade and A. Ghotkar. Animation of 3d human model using markerless motion capture applied to sports. *International Journal of Computer Graphics & Animation*, 4(1).

[100] Skeltrack - reference manual. Accessed: 26-08-2015. URL: `http://people.igalia.com/jrocha/skeltrack/doc/latest/`.

[101] Summit medical group - adult health advisor. Accessed: 30-08-2015. URL: `http://www.summitmedicalgroup.com/library/adult_health/`.

[102] E. Trucco and A. Verri. *Introductory Techniques for 3-D Computer Vision*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1998.

[103] H. Hirschmuller. Stereo processing by semiglobal matching and mutual information. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 30(2):328–341, Feb 2008.

[104] K. Konolige. Small vision systems: Hardware and implementation. In *Robotics Research*, pages 203–212. Springer London, 1998.

[105] S Birchfield and C. Tomasi. A pixel dissimilarity measure that is insensitive to image sampling. *IEEE Trans. Pattern Anal. Mach. Intell.*, 20(4):401–406, April 1998.

[106] F. Hu and Y. Zhao. Comparative research of matching algorithms for stereo vision. *Journal of Computational Information Systems*, 9(11):5457–5465, 2013.

[107] S. Kosov, T. Thormählen, and H. P. Seidel. Accurate real-time disparity estimation with variational methods. In *Advances in Visual Computing*, volume 5875 of *Lecture Notes in Computer Science*, pages 796–807. Springer Berlin Heidelberg, 2009.

[108] Opencv 2.4.9.0 documentation: Stereo correspondence. Accessed: 29-01-2015. URL: `http://docs.opencv.org/modules/contrib/doc/stereo.html`.

[109] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision*, 47(1-3):7–42, 2002.

[110] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004.

[111] D. Gu, Y. Zhao, Y. Yuan, and G. Hu. Human segmentation based on disparity map and grabcut. pages 67–71, Dec 2012.

[112] C. Rother, V. Kolmogorov, and A. Blake. Grabcut -interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics (SIGGRAPH)*, August 2004.

[113] N. Otsu. A Threshold Selection Method from Gray-level Histograms. *IEEE Transactions on Systems, Man and Cybernetics*, 9(1):62–66, 1979.

[114] M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Commun. ACM*, 24(6):381–395, 1981.

[115] R. B. Rusu, Z. C. Marton, N. Blodow, M. Dolha, and M. Beetz. Towards 3d point cloud based object maps for household environments. *Robotics and Autonomous Systems*, 56(11):927 – 941, 2008.

[116] S. Paris and F. Durand. A fast approximation of the bilateral filter using a signal processing approach. *International Journal of Computer Vision*, 81(1):24–52, 2009.

[117] F. Durand and J. Dorsey. Fast bilateral filtering for the display of high-dynamic-range images. *ACM transactions on graphics (TOG)*, 21(3):257–266, 2002.

[118] K. Buys, C. Cagniart, A. Baksheev, T. D. Laet, J. D. Schutter, and C. Pantofaru. An adaptable system for rgb-d based human body detection and pose estimation. *Journal of Visual Communication and Image Representation*, 25(1):39 – 52, 2014.

[119] Openni. Accessed: 29-08-2015. URL: `http://structure.io/openni`.

[120] Pcl gpu people. Accessed: 29-08-2015. URL: `http://docs.pointclouds.org/trunk/namespacepcl_1_1gpu_1_1people.html`.

[121] K. Buys, D.V. Deun, T.D. Laet, and H. Bruyninckx. On-line generation of customized human models based on camera measurements. *International Symposium on Digital Human Modeling*, 2011.

[122] D.V. Deun, V. Verhaert, K. Buys, B. Haex, and J.V. Sloten. Automatic generation of personalized human models based on body measurements. *International Symposium on Digital Human Modeling*, 2011.

[123] J. Dean and S. Ghemawat. Mapreduce: Simplified data processing on large clusters. *Commun. ACM*, 51(1):107–113, 2008.

[124] Pcl developers blog - alina roitberg. Accessed:15-08-2015. URL: `http://pointclouds.org/blog/gsoc14/aroitberg/index.php`.

[125] M. Munaro, F. Basso, and E. Menegatti. Tracking people within groups with rgb-d data. In *Intelligent Robots and Systems (IROS), 2012 IEEE/RSJ International Conference on*, pages 2101–2107, Oct 2012.

[126] M. Munaro and E. Menegatti. Fast rgb-d people tracking for service robots. *Autonomous Robots*, 37(3):227–242, 2014.

[127] R. Drillis, R. Contini, and M. Bluestein. Body segment parameters; a survey of measurements and techniques. *Artif Limbs*, 8:44–66, 1964.

[128] R. E. Kalman. A New Approach to Linear Filtering and Prediction Problems. *Transactions of the ASME – Journal of Basic Engineering*, (82 (Series D)):35–45, 1960.

[129] S. Jun, X. Zhou, D. K. Ramsey, and V. N. Krovi. A comparative study of human motion capture and analysis tools a comparative study of human motion capture and computational analysis tools. *International Conference on Rehabilitation Robotics*, pages 1–8, 2013.

[130] P. Soltani and J.P. Vilas-Boas. *Muscle activation during exergame playing.* IGI Global.

[131] J. Shotton, R. Girshick, A. Fitzgibbon, T. Sharp, M. Cook, M. Finocchio, R. Moore, P. Kohli, A. Criminisi, A. Kipman, and A. Blake. Efficient human pose estimation from single depth images. *Trans. PAMI*, 2012.

[132] Anders Boesen Lindbo Larsen, Søren Hauberg, and KimSteenstrup Pedersen. Unscented kalman filtering for articulated human tracking. In *Image Analysis*, volume 6688 of *Lecture Notes in Computer Science*, pages 228–237. Springer Berlin Heidelberg, 2011.

[133] M. Windolf, N. Götzen, and M. Morlock. Systematic accuracy and precision analysis of video motion capturing systems—exemplified on the vicon-460 system. *Journal of Biomechanics*, 41(12):2776 – 2780, 2008.

[134] W. R. Taylor, R. M. Ehrig, G. N. Duda, H. Schell, P. Seebeck, and M. O. Heller. On the influence of soft tissue coverage in the determination of bone kinematics using skin markers. *Journal of Orthopaedic Research*, 23(4):726–734, 2005.

[135] J. Canny. A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 8(6):679–698, Nov 1986.